



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA PODNIKATELSKÁ
ÚSTAV INFORMATIKY**

FACULTY OF BUSINESS AND MANAGEMENT
INSTITUTE OF INFORMATICS

NÁVRH SYSTÉMU PRO ZPRACOVÁNÍ DAT HODNOCENÍ ZAMĚSTNANCŮ

DESIGN OF A SYSTEM FOR PROCESSING EMPLOYEE ASSESSMENT DATA

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

FILIP BOŠTÍK

VEDOUcí PRÁCE

SUPERVISOR

Ing. PETR DYDOWICZ, Ph.D.

BRNO 2015

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Boštík Filip

Manažerská informatika (6209R021)

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách, Studijním a zkušebním řádem VUT v Brně a Směrnicí děkana pro realizaci bakalářských a magisterských studijních programů zadává bakalářskou práci s názvem:

Návrh systému pro zpracování dat hodnocení zaměstnanců

v anglickém jazyce:

Design of a System for Processing Employee Assessment Data

Pokyny pro vypracování:

Úvod

Vymezení problému a cíle práce

Teoretická východiska práce

Analýza problému a současné situace

Vlastní návrh řešení, přínos práce

Závěr

Seznam použité literatury

Seznam odborné literatury:

BRADEN, Melanie a Michael SCHWIMMER. Excel 2007 VBA. Velká kniha řešení. Brno: Computer Press, a.s., 2009. 685 s. ISBN 978-80-251-2698-1.

ČIHAŘ, Jiří. 1001 tipů a triků pro Microsoft Excel 2007/2010. Brno: Computer Press, a.s., 2011. 488 s. ISBN: 978-80-251-2587-8.

KRÁL, Martin. Excel VBA. Výukový kurz. Brno: Computer Press, a.s., 2010. 504 s. ISBN 978-80-251-2358-4.

KRÁL, Mojmír. Excel 2010 – snadno a rychle. Praha: Grada Publishing a.s., 2010. 143 s. ISBN 80-2473-495-8.

LAURENČÍK, Marek. Programování v Excelu 2007 a 2010. Praha: Grada Publishing a.s., 2011. 192 s. ISBN 978-80-247-3448-4.

Vedoucí bakalářské práce: Ing. Petr Dydowicz, Ph.D.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2014/2015.

L.S.

doc. RNDr. Bedřich Půža, CSc.
Ředitel ústavu

doc. Ing. et Ing. Stanislav Škapa, Ph.D.
Děkan fakulty

V Brně, dne 28.2.2015

Abstrakt

Tato bakalářská práce se zabývá návrhem systému pro hromadné zpracovávání dat hodnocení zaměstnanců ve firmě IBM Global Services Delivery Center Czech Republic. Vstupní data jsou poskytována zaměstnanci oddělení ve formě odpovědí na otázky dotazníkového typu, výstupem systému je analýza dat pro vedení oddělení. V první kapitole práce je popsána teorie důležitá pro pochopení práce, a ve druhé části je pak problém analyzován. V poslední kapitole je proveden výběr a popis navrhovaného systému.

Abstract

This thesis describes the design of a system for batch data processing of employee assessment data in company IBM Global Services Delivery Centre Czech Republic. Input data is provided by employees of the department in the form of answers to questions of the questionnaire type, output of the system is an analysis for the management of the department. The first chapter describes the theory important for understanding the thesis, and the second chapter contains the analysis of the problem. In the last chapter there is the selection and an overview of the designed system.

Klíčová slova

Hodnocení, zaměstnanci, Microsoft Office, Excel, IBM, Bluemix.

Keywords

Assessment, employee, Microsoft Office, Excel, IBM, Bluemix.

Bibliografická citace

BOŠTÍK, F. *Návrh systému pro zpracování dat hodnocení zaměstnanců*. Brno: Vysoké učení technické v Brně, Fakulta podnikatelská, 2015. 76 s. Vedoucí bakalářské práce Ing. Petr Dydowicz, Ph.D..

Čestné prohlášení

Prohlašuji, že předložená bakalářská práce je původní a zpracoval jsem ji samostatně. Prohlašuji, že citace použitých pramenů je úplná, že jsem ve své práci neporušil autorská práva (ve smyslu Zákona č. 121/2000 Sb., o právu autorském a o právech souvisejících s právem autorským).

V Brně dne 26. května 2015

.....

Poděkování

Tímto bych chtěl poděkovat lidem, bez kterých by tato práce nemohla vzniknout. Nejdříve bych chtěl poděkovat vedení oddělení IGA EMEA Projects and Programmes za umožnění tvorby této práce. Dále bych chtěl poděkovat všem členům týmu Project Office a ostatním zaměstnancům oddělení za pomoc při získávání informací. Poslední dík patří panu Ing. Petru Dydowiczovi, Ph.D., za to, že moji práci vedl.

OBSAH

ÚVOD.....	11
VYMEZENÍ PROBLÉMU A CÍLE PRÁCE	12
1 TEORETICKÁ VÝCHODISKA PRÁCE	13
1.1 Hodnocení zaměstnanců.....	13
1.1.1 Motivace a stimulace	13
1.2 Visual Basic for Applications	14
1.3 Databáze	15
1.3.1 Relační databáze	15
1.3.2 SQL.....	16
1.3.3 NoSQL databáze	17
1.4 Cloud	17
1.4.1 Infrastructure as a Service (IaaS).....	17
1.4.2 Platform as a Service (PaaS).....	17
1.4.3 Software as a Service (SaaS)	17
1.5 Informační systém.....	18
1.6 Některé další pojmy z oblasti Informačních technologií	18
1.6.1 Kontingenční tabulky.....	18
1.6.2 Pojmy z oblasti vývoje softwaru.....	18
1.6.3 Big Data	19
1.7 Analýza HOS 8	19
1.8 Použité diagramy.....	20
1.8.1 Vývojový diagram	20
1.8.2 Diagram toku dat	21
2 ANALÝZA PROBLÉMU A SOUČASNÉ SITUACE	22
2.1 Stručný popis vybrané společnosti a oddělení	22
2.1.1 IBM.....	22

2.1.2	IBM Global Services Delivery Center Czech Republic	23
2.1.3	IBM Global Account	24
2.1.4	Oddělení IGA EMEA Projects and Programs	24
2.2	Popis řešeného problému	25
2.2.1	Původní stav formulářů.....	26
2.2.2	Požadavky na systém	27
2.2.3	Stanovení základních možností řešení	28
2.3	Analýza současného stavu hardwaru	28
2.3.1	Pracovní stanice	28
2.3.2	Počítačová síť a Servery	29
2.4	Analýza současného stavu softwaru.....	29
2.4.1	Kancelářské balíky.....	29
2.4.2	IBM Connections	32
2.4.3	IBM Program Work Center	33
2.4.4	IBM Bluemix	35
2.4.5	Vybrané technologie pro vývoj webových aplikací	36
2.5	Souhrn analýzy.....	37
3	VLASTNÍ NÁVRH ŘEŠENÍ, PŘÍNOS PRÁCE	39
3.1	Možnosti řešení	39
3.1.1	Implementace do IPWC.....	39
3.1.2	Webová aplikace.....	40
3.1.3	Systém založený na aplikaci Microsoft Excel a IBM Connections.....	41
3.1.4	Výběr řešení	42
3.2	Popis vybraného řešení	43
3.2.1	Postup provádění hodnocení a analýzy.....	43
3.2.2	Stanovení výstupních atributů hodnocení.....	44
3.3	Návrh formuláře	45

3.3.1	Ošetření vstupních hodnot	46
3.3.2	Dodatečné úpravy formuláře	47
3.4	Návrh aplikace na zpracování formulářů	48
3.4.1	Podpůrné podprogramy.....	49
3.4.2	Komunikace programu s uživatelem	50
3.4.3	Import dat.....	52
3.4.4	Export dat.....	54
3.4.5	Tvorba grafů	58
3.4.6	Nastavení programu	62
3.4.7	Výběr formátu aplikace	63
3.5	Návrh komunity v IBM Connections	63
3.6	Webová aplikace postavená na platformě IBM Bluemix	64
3.6.1	Stručný popis aplikace	64
3.6.2	Ekonomické zhodnocení aplikace postavené na IBM Bluemix	68
3.7	Ekonomické zhodnocení navrhovaného řešení	69
3.7.1	Náklady	69
3.7.2	Přínosy	69
ZÁVĚR		70
SEZNAM POUŽITÉ LITERATURY		71
SEZNAM POUŽITÝCH ZKRATEK.....		74
SEZNAM OBRÁZKŮ.....		75
SEZNAM DIAGRAMŮ		75
SEZNAM GRAFŮ		75
SEZNAM TABULEK		76
SEZNAM KÓDU		76

ÚVOD

Nejdražším zdrojem snad všech podniků jsou jejich zaměstnanci. Firmy by bez tohoto zdroje ani nemohly existovat, a proto je důležité znát jeho kvalitu. Mnoho větších firem tedy používá různé systémy hodnocení zaměstnanců, ať už zakoupené od renomovaných výrobců informačních systémů nebo vzniklé vlastní činností, aby mohly objektivnějším způsobem zhodnotit a poté i efektivněji řídit tento důležitý zdroj.

V této práci jsem se tedy rozhodl popsat tvorbu technické části takového systému, pro konkrétní oddělení konkrétní společnosti. Společnost IBM (i její dceřiné společnosti) sice podobná řešení mají, ale vedení oddělení IGA EMEA Projects and Programmes chtělo vytvořit systém, zaměřený přesně na znalosti, zkušenosti a vlastnosti zaměstnanců, které jsou pro hladké fungování tohoto oddělení třeba a podpořit tak stávající procesy hodnocení zaměstnanců.

Na následujících stranách tedy, po teorii nezbytné pro pochopení této práce a analýze současného stavu oddělení a dostupných technologií, popíši některá řešení, která jsem pro navrhovaný systém zvažoval, a poté se zaměřím na řešení, které jsem pro oddělení vytvořil.

VYMEZENÍ PROBLÉMU A CÍLE PRÁCE

Cílem této bakalářské práce je návrh systému pro hromadné zpracovávání dat hodnocení zaměstnanců pro oddělení IGA EMEA Projects and Programmes, společnosti IBM. Vstupní data pro tento systém budou poskytována ve formě odpovědí na dotazníkové otázky vybranými zaměstnanci. Systém má pomoci zlepšit představu vedení oddělení o zkušenostech a schopnostech zaměstnanců.

V první části práce rozeberu základní teorii potřebnou k pochopení této práce, jako například krátký úvod do hodnocení zaměstnanců, základy tvorby relačních databází nebo několik důležitých pojmů z oblasti návrhu softwaru.

Pro návrh jakéhokoli systému je potřeba důkladné pochopení daného problému a následně i možností jeho řešení. Tím se budu zabývat v druhé kapitole, kde nejprve popíši současnou organizační strukturu zkoumaného oddělení. Pokračovat budu stavem výpočetní techniky, dostupného hardwarového vybavení a běžně dostupného softwaru. Pro jeden ze systémů oddělení provedu analýzu pomocí metody HOS 8. Rovněž stanovím požadavky na daný systém, které mi následně umožní výběr vhodného řešení a platformy.

Výběrem vhodného řešení budu pokračovat v třetí části této práce. Nastíním možná řešení, zhodnotím jejich možné přínosy a úskalí jejich tvorby. Následně vyberu nejvhodnější možnost, na kterou se poté zaměřím. Ukáži logiku a základní části systému a nastíním i organizační část tohoto systému. V závěru této kapitoly také zhodnotím ekonomické přínosy a náklady daného řešení.

1 TEORETICKÁ VÝCHODISKA PRÁCE

1.1 Hodnocení zaměstnanců

Pod tímto pojmem rozumějme soubor různých metod, které mají za úkol vyjádřit výkonnost a chování zaměstnanců. Hodnocení zaměstnanců spadá do oboru personalistiky a zpravidla je prováděno manažery nebo personalisty. Mezi hlavní důvody, proč se hodnocení zaměstnanců provádí, obvykle patří zvýšení výkonnosti, díky zpětné vazbě poskytnuté zaměstnancům, a zlepšení řízení dané organizační jednotky. Dále pak může vést ke sladování zájmu organizace a posílení nebo změně podnikové kultury.

Zaměstnanci by měli být hodnoceni ve třech hlavních kategoriích. Výstup neboli výkon (odvedená práce) a vstup, což značí věci jako vzdělání nebo odborná praxe. Poslední kategorií je poté proces, do které spadá hlavně přístup zaměstnanců.

Také existuje mnoho kritérií, podle kterých se dá hodnocení zaměstnanců dělit. Například na spontánní a organizované, nebo například hodnocení dle dovedností, popřípadě schopností pracovníka. Subjektivní hodnocení je většinou ústní, a nejčastěji k němu dochází při mimořádných situacích. Organizované hodnocení bývá většinou písemné a provádí se v pravidelných intervalech (1).

1.1.1 Motivace a stimulace

Motivace a stimulace jsou dva podobné pojmy. Oba představují vliv na výkonnost pracovníka. Rozdílem je ale směr tohoto vlivu. Stimulace značí vliv z okolí zaměstnance a nejčastěji se jí rozumějí peníze. Motivace je naopak vnitřní vliv, kterým může být například přání člověka dosáhnout určitého cíle.

Hodnocení pracovníků může být jak motivačním, tak i stimulačním prvkem. O motivaci můžeme mluvit, pokud hodnocený zaměstnanec hodnocení použije jako zpětnou vazbu k odváděné práci a bude usilovat o zlepšení. Pokud bude hodnocení využito k odměňování či „trestání“ (metoda „Cukr a bič“), pak se bude jednat o stimulaci (2).

1.2 Visual Basic for Applications

Visual Basic for Applications (VBA), je událostním, objektově orientovaným jazykem vycházejícím z jazyku Visual Basic v.6 od společnosti Microsoft. S různými objektovými strukturami je VBA integrováno snad do všech aplikací kancelářského balíku Microsoft Office. Při použití VBA se použitá aplikace (například Excel) stává aplikací hostitelskou, a kód (často označovaný jako makra) nad ní přebírá plnou kontrolu (3).

V následující části popíši několik důležitých pojmů programování a jejich vliv na fungování tohoto programovacího jazyka.

Procedury a funkce

Procedury a funkce chápeme jako pojmenovaný kus kódu, který lze spouštět z jiného místa programu, pomocí tohoto jména. Procedury provedou zadaný kód, ale nevrací žádnou hodnotu. Hlavním účelem funkcí je zjištění a vrácení určité hodnoty. V jiných programovacích jazycích pro procedury a funkce existují souhrnné názvy, jako jsou podprogramy nebo metody (4).

Události

Události představují určitou akci programu, většinou způsobenou uživatelem, která může spustit jí přiřazený podprogram. Mezi běžné události ve VBA patří například stisknutí tlačítka, otevření sešitu nebo změna dat ve vybraném listu (4).

Objektově orientované programování

Objektově orientované programování stojí na principu používání tzv. objektů, které jsou vytvořeny podle určitého předpisu (třídy). Třidu tedy chápeme jako plán nebo předpis, podle kterého jsou tvořeny objekty, které mohou nabývat různých hodnot, a dále pak fungovat zcela nezávisle na ostatních objektech stejné třídy. Třídy deklarují vlastnosti (proměnné), které objekt bude mít, a procedury a funkce, které bude moci používat. Na objekty se po deklaraci odkazujeme pomocí referencí (odkazů), což je zvláštní druh proměnné.

Objekty v aplikaci Excel představují například samotnou aplikaci, sešit nebo list aplikace, ale také například buňku, do které můžeme zapisovat (4).

Moduly

Ve VBA moduly označují tzv. knihovny, což je v podstatě kód, uložený do samostatného souboru. Dělí se na moduly a moduly tříd. Rozdíl je ten, že moduly tříd by měly ukládat pouze kód, vztahující se k deklaraci jedné třídy. „Běžné“ moduly mohou ukládat například programy (4).

1.3 Databáze

Pojem databáze je nejčastěji vysvětlován jako „organizovaná množina dat“. Často se dělí podle datových modelů. Mezi tyto modely patří například lineární, síťový nebo hierarchický model. Dále se pak v databázích používá relační nebo objektový model.

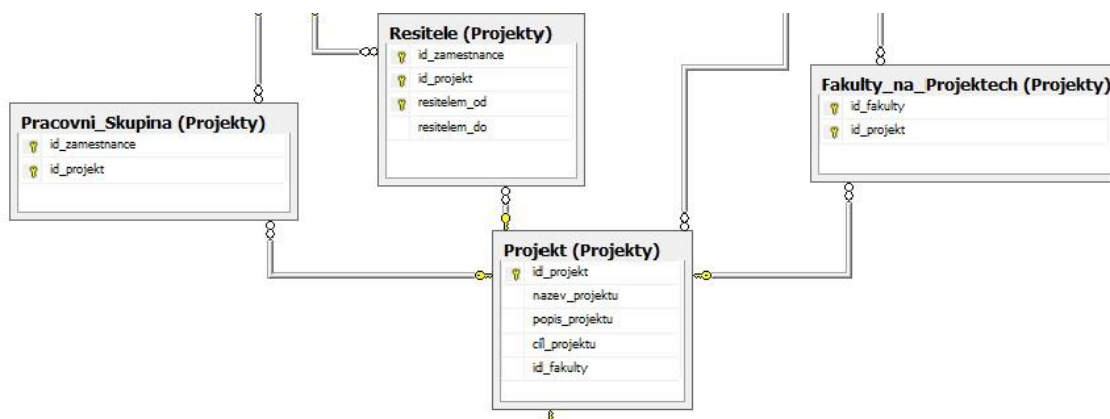
V dnešní době jsou databáze snad nejčastěji implementované v podobě relačních databází (relační datový model), uložených v databázových aplikacích, a pomocí jazyků odvozených od jazyka SQL. Dalším rozšířeným přístupem k databázím jsou NoSQL databáze (5).

1.3.1 Relační databáze

Relační databáze jsou založené na tabulkách a jejich spojeních. Tabulky neboli relace, se skládají ze sloupců a řádků. Řádky obsahují hodnoty, které se týkají jednoho záznamu, tzv. n-tice relace (například studenta nebo výrobku), zatímco sloupce obsahují hodnoty jednoho druhu a datového typu - atribut (například jména, nebo věk studenta, rozměry výrobku apod.). Pro snadnou identifikaci jednotlivých záznamů se do tabulek zavádí tzv. primární klíč. Ten jednoznačně identifikuje jednotlivé záznamy a často je tvořen samostatným záznamem, nazvaným „ID“.

V relačních databázích je možné tabulky spojovat. Pro spojení dvou tabulek potřebujeme dva klíče. Prvním je již zmíněný primární klíč a tím druhým je tzv. cizí klíč. Opět se jedná o samostatný sloupec v tabulce, který má odpovídající sloupec v druhé tabulce, kterou chceme připojit. V této druhé tabulce je cizí klíč z první tabulky primárním klíčem v tabulce druhé.

Spojením více tabulek nám vzniká schéma databáze, které se velmi často zobrazuje graficky (pomocí tzv. ER diagramu). Schéma databáze si můžeme zjednodušeně představit jako plán, nebo mapu konkrétní databáze (5).



Obrázek č. 1: Příklad databázového schéma ve formě ER diagramu (vlastní zpracování)

Základní principy návrhu relační databáze

Při tvorbě relační databáze bychom měli dodržovat určitá pravidla, aby databáze pracovala co nejefektivněji a neobsahovala zbytečná (redundantní) data. K tomu nám pomůže tzv. normalizace. Skládá se z několika pravidel, přičemž uvedu ty nejdůležitější.

- 1. normální forma (multizávislost): v relaci se nesmí vyskytovat žádný atribut, který by nebyl atomární nebo by byl vícehodnotový.
- 2. normální forma (funkční závislost): každý z atributů relace musí být závislý na celém primárním klíči.
- 3. normální forma (tranzitivní závislost): žádné atributy v tabulce nesmí být zaměnitelné (5).

Při návrhu vazeb mezi tabulkami je rovněž potřeba brát v potaz tzv. kardinalitu vazeb. Ta značí, kolikrát se může prvek (řádek) z jedné tabulky, vyskytnou v druhé tabulce a naopak. V případě, že se nám v tabulce vyskytne vazba, která značí více hodnot na obou stranách tabulky (často značená jako „N:M“), pak tuto vazbu musíme dekomponovat, na dvě vazby 1:N. K tomu použijeme novou tabulku (5).

1.3.2 SQL

SQL (Structured Query Language) je jazyk pro komunikaci s databází. Umožňuje uživateli činnosti, jako je tvorba tabulek, vkládání nových dat do databáze nebo jejich výpis a úpravy. SQL také umožňuje tvořit procedury (podprogramy) a pohledy („virtuální tabulky“). V dnešní době je používán s nejrůznějšími databázovými systémy, jako je MySQL, DB2, Microsoft SQL atd. (6).

1.3.3 NoSQL databáze

Jedná se o skupinu databázových systémů většinou pracujících na principu dat uložení dat ve formátu klíč-hodnota. Jejich největší odlišností od relačních databází (a i dalších databázových modelů) je, že nepoužívají pevně danou strukturu dat (například tabulky). Existuje několik různých implementací, jako je Dynamite, Hypertable, Apache HBase nebo mongoDB. NoSQL databáze se využívají hlavně pro ukládání dat pro analýzy „velkých dat“ nebo pro některé webové aplikace (7).

1.4 Cloud

Pojem „cloud“ (nebo „cloud computing“) v sobě zahrnuje pronájem softwaru nebo výpočetní kapacity ve formě služby. K dané službě se poté přistupuje přes internet (většinou přes internetový prohlížeč). K tomuto pojmu se ovšem váže několik dalších termínů (8).

1.4.1 Infrastructure as a Service (IaaS)

Značí pronájem infrastruktury, tudíž například virtualizovaných operačních systémů nebo datových uložišť. Uživatel (nájemce) tudíž nespravuje fyzická zařízení (hardware), ale využívá například virtuální operační systém, na kterém může nechat běžet své aplikace a běhová prostředí (9).

1.4.2 Platform as a Service (PaaS)

Poskytování platformy chápeme, jako poskytování nejrozličnějších běhových prostředí, nebo jiných služeb pro běh webových a mobilních aplikací. Výhodou tohoto modelu je možnost umístění vlastních aplikací, bez nutnosti správy samotného prostředí, ve kterém aplikace poběží (9).

1.4.3 Software as a Service (SaaS)

Jedná se o pronájem již hotových aplikací, velmi často se jedná o informační systémy, sociální sítě, nebo emailové stránky. Jednou z hlavních výhod modelu SaaS je, že za funkčnost samotné aplikace zodpovídá poskytovatel (8).

1.5 Informační systém

Pro tento pojem existuje mnoho nejrůznějších definic. Jedna z nich jej definuje jako „...integrovaný soubor komponent pro sběr, uložení a zpracování dat a pro doručování informací znalostí a digitálních produktů...“ (10). Tuto definici bychom v dnešní době mohli rozšířit o použití moderních informačních technologií, jako jsou počítače, databáze nebo cloud.

1.6 Některé další pojmy z oblasti Informačních technologií

1.6.1 Kontingenční tabulky

Kontingenční tabulka je interaktivním nástrojem pro zpracování dat, který se často integruje do tabulkových procesorů. Takováto aplikace zpracuje zadaná numerická data podle zadaných požadavků a vytvoří jejich přehled, přičemž se většinou jedná o součty nebo průměry dat se společnými parametry (například průměrné tržby v pobočkách obchodu ve sledovaném období). Další funkcionalitou, kterou tyto tabulky nabízejí je třídění a filtrování zobrazených dat. Na stejném principu také fungují kontingenční grafy, které, jak napovídá název, tvoří grafickou reprezentaci dat zobrazených v kontingenční tabulce (11).

1.6.2 Pojmy z oblasti vývoje softwaru

Běhové prostředí (Runtime, Runtime system) je program, který poskytuje běžícímu programu určité služby. Implementace Běhového prostředí se liší v závislosti na konkrétním jazyce a jeho vlastnostech. Běhová prostředí tedy mohou zajišťovat například vstupně výstupní operace, komunikaci s operačním systémem, zpracování chyb nebo volání podprogramů z jiného programovacího jazyka (12).

API (Application Programming Interface – Aplikační programové rozhraní) je programový kód, který umožňuje rozšíření programu o jím definovanou funkcionalitu. Hlavním účelem API je rychlejší vývoj software, protože programátor již nemusí tvořit kód pro danou funkcionalitu, ale pouze do programu přidá dané API a jeho volání. V různých programovacích jazycích můžeme nalézt různé názvy pro API, jako například „knihovny“ v jazyce C, nebo „moduly“ ve VBA (13).

Framework (aplikační rámec) chápeme jako platformu pro vývoj aplikací pro určitý operační systém. Může poskytovat kód pro operace spojené s hardwarovými zařízeními nebo pro komunikaci s operačním systémem. Od API se Framework liší především tím, že API poskytuje základní funkce, některé programy užívané při vývoji SW (například kompilátor), ale hlavně, že může obsahovat více API (14).

Software development kit (SDK) představuje nástroje pro psaní programů pro určitý Framework (14).

1.6.3 Big Data

Tzv. velkými daty myslíme velké objemy strukturovaných i nestrukturovaných dat, které se zpracovávají a analyzují pro nejrůznější účely od zpracování firemních dat, údajů o počasí nebo i dat ve zdravotnictví (15).

DevOps

DevOps pochází z anglického Development and Operation (Vývoj a Provoz). Jedná se o přístup k vývoji a provozu software, který má zlepšovat spokojenost zákazníka, a zlepšovat kvalitu a spolehlivost vyvíjeného softwaru. V současné době se jedná spíše o přístup, než přesně definovanou metodiku. Proto také neexistuje jeden standartní nástroj, ale mnoho aplikací tohoto přístupu od různých společností (16).

1.7 Analýza HOS 8

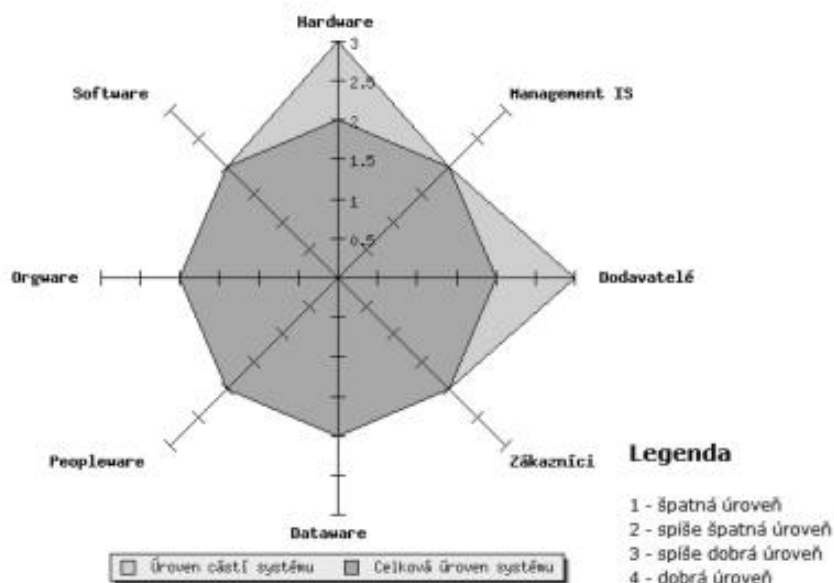
Analýza HOS je metodou sloužící ke zjištění stavu informačního systému. Existuje několik verzí této metody, v této práci použijí verzi HOS 8, kterou nyní popíši.

Metoda HOS 8 je založena na odpovědích na otázky z osmi oblastí informačních systémů (hardware, software, orgware, peopleware, dataware, zákazníci, dodavatelé, management IS) a nejnovější verze metody také posuzuje také bezpečnost tohoto systému. Na každou otázku existují čtyři nebo pět odpovědí (podle verze analýzy), které mají přiřazenou číselnou hodnotu, kterou ale hodnotící nezná. Tato hodnota poté slouží k následnému vyhodnocení každé z oblastí analýzy. Vyhodnocením výsledků analýzy vznikne graf, zobrazující stav částí IS (stejných, jako jsou okruhy otázek), a také stav celého IS. Celkovým stavem IS je pak dosažená úroveň nejhůře hodnocené části

zkoumaného systému. Dosaženou hodnotu je pak vhodné porovnat vzhledem k důležitosti IS pro organizaci a podle toho také nastavit další informační strategii.

Hlavním omezením metody je její jednoduchost (nehodnotí totiž jednotlivé procesy, ale pouze hlavní části IS), a také fakt, že metoda je založena na subjektivních hodnocení respondenta (17).

Analýzu HOS 8 je možné provést také pomocí internetového portálu Zefis.cz (18).



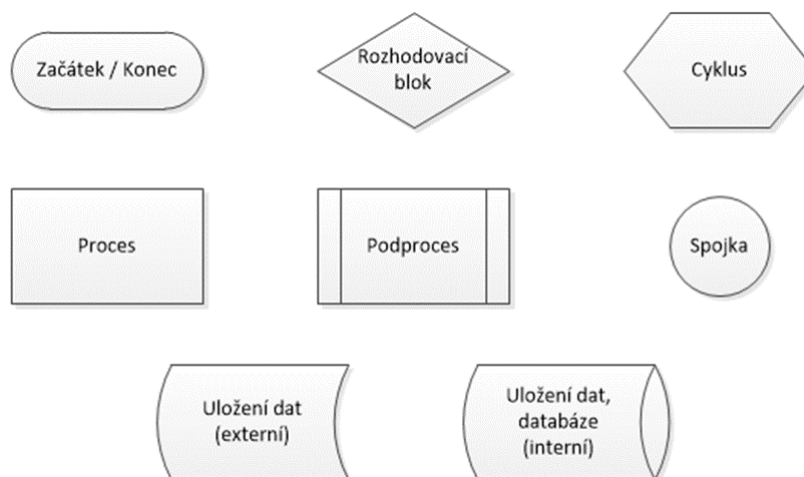
Obrázek č. 2: Ukázka výstupu analýzy HOS 8 (17)

1.8 Použité diagramy

V následující kapitole popíši diagramy, které budu používat v této práci.

1.8.1 Vývojový diagram

Vývojový diagram umožňuje sledování logického větvení zkoumané úlohy. Vývojové diagramy se často používají k zachycení algoritmů uvnitř počítačových programů. Symboly zde představují činnosti, které by v programu byly zachycené kódem. Návaznost jednotlivých symbolů je dána šipkami a měl by se dodržovat základní tok úlohy, což je shora dolů a zleva doprava. Ve vývojových diagramech se používá více symbolů, já zde ale uvedu pouze ty nejdůležitější (5).



Obrázek č. 3: Některé symboly používané ve vývojových diagramech (vlastní zpracování podle: 5)

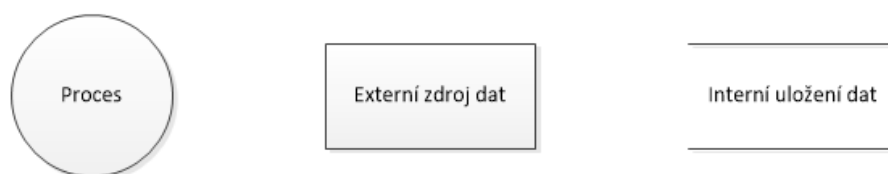
Symbol podprocesu značí, že daný proces bude dále popsán, buď textem, nebo dalším diagramem. Spojka se používá pro zlepšení přehlednosti diagramů a značí přeskok na další, stejně značenou spojku. Spojky se obvykle číslují (5).

1.8.2 Diagram toku dat

Diagram toku dat neboli Data Flow Diagram (DFD) patří mezi diagramy používané ve funkčním modelování. Jak již napovídá název, jeho hlavním úkolem je zachytit tok dat ve zkoumané úloze. Často se diagram nejprve vytvoří pro základní prvky úlohy a následně se jednotlivé části úlohy dále rozkládají.

V DFD diagramech se používají tři druhy symbolů. „Proces“ značí práci s daty, „externí zdroj“ dat pak značí vstup dat ze zdroje mimo zkoumaný systém. Poslední značka je „uložení dat uvnitř systému“. Dále se pak ještě používají šipky, které znázorňují vazby mezi symboly.

Pro DFD existují tři běžně užívané sady symbolů (notace): Gane and Sarson, Yourdon and Coad a SSDAM. V této práci budu používat notaci Yourdon and Coad (5).



Obrázek č. 4: Značky DFD podle notace Yourdon and Coad (vlastní zpracování podle: 5)

2 ANALÝZA PROBLÉMU A SOUČASNÉ SITUACE

V následující kapitole se zaměřím na současný stav oddělení IGA EMEA Projects and Programmes a jeho informačních technologií. Také popíši požadavky, které budu klást na systém navrhovaný v této práci.

2.1 Stručný popis vybrané společnosti a oddělení

Jak již bylo naznačeno, kapitolu začnu popisem společnosti IBM a jejího oddělení IGA EMEA Projects and Programmes. Všechny následující údaje jsou aktuální k lednu roku 2015.

2.1.1 IBM

International Business Machines Corporation nebo zkráceně IBM je americká akciová společnost. Společnost působí na trhu již od roku 1911, přičemž pod současným jménem působí od roku 1924. Z počátku se společnost zabývala převážně zařízeními pro zpracování dřevných štítků, hodin atd., a od 40. let se společnost zabývá i výrobou počítačů. Asi nejznámějším produktem společnosti je IBM PC (19).

IBM v současné době operuje po celém světě a nabízí hardware (mainframe, servery, uložení atd.), software (WebSphere, Cognos, Tivoli atd.) a služby (outsourcing, cloud atd.) (20).

Tomu také odpovídá rozdělení do základních segmentů, které se dále dělí na obchodní skupiny (divize).

- Services Segments (Segment služeb).
 - Global Business Services (Globální obchodní služby).
 - Global Financing (Globální financování).
 - Global Technology Services (Globální technické služby).
- Software Segment (Segment softwaru).
- Systems & Technology Segment (Segment systémů a technologií) (20).

IBM se také dělí dle „lokací“:

- Americas – USA, Kanada a Jižní Amerika.
- AP – Japonsko, Asie a Pacifik.
- EMEA – Evropa, Střední východ, Afrika (20).

Vize společnosti je definována následovně:

Být pro klienty nezbytným partnerem jejich byznysu.

- Zaujetí úspěchem našich klientů - jejich úspěch je náš úspěch.
- Inovace, které mění svět.
- Důvěra a osobní zodpovědnost ve všech vztazích (21).

Velice důležitou částí současné strategie IBM je tzv. Smarter Planet, neboli „Chytřejší planeta“. Jedná se o soubor projektů, iniciativ a produktů, které mají za úkol zvýšit využití informačních technologií za účelem zmenšení spotřeby zdrojů, zvýšení efektivnosti lékařské péče nebo dopravy apod. IBM tímto nemíří pouze na soukromý sektor, ale také na veřejný sektor (20).

Mezi největší konkurenty IBM můžeme řadit firmy jako: HP, Amazon, Microsoft, Oracle. Vždy ovšem záleží na konkrétním produktu a jeho zaměření.

Základní údaje:

- Sídlo: New Orchard Road, Armonk, New York 10504.
- Počet zaměstnanců: 431 tisíc (leden 2015).
- Prezident a CEO: Virginia M. Rometty (20).

2.1.2 IBM Global Services Delivery Center Czech Republic

Jedná se o dceřinou společnost IBM, sídlící v Brně. Spadá pod IBM Global Services (Segment Služeb) a jedná se o tzv. Delivery Center (DC). DC mají za úkol především poskytování služeb zákazníkům společnosti všude po světě. Nejdůležitější činnosti brněnské pobočky se zaměřují na strategický outsourcing, tudíž na podporu serverů, aplikací, síťových služeb a koncových klientů. Zákazníci brněnské pobočky IBM se nacházejí všude po světě, přičemž se většinou jedná o střední a velké podniky (22).

Základní údaje:

Obchodní firma: IBM Global Services Delivery Center Czech Republic, s.r.o.

Sídlo: Technická 2995/21, Královo Pole, 616 00 Brno.

Právní forma: Společnost s ručením omezeným.

Předmět podnikání: výroba, obchod a služby neuvedené v přílohách 1 až 3 živnostenského zákona, zprostředkování zaměstnání (23).

2.1.3 IBM Global Account

Tuto práci zpracovávám pro IBM Global Account (zkráceně IGA), přesněji pro oddělení IGA EMEA Projects and Programmes (P&P). Hlavním činností IGA je poskytování IT služeb a technického zázemí ostatním částem IBM.

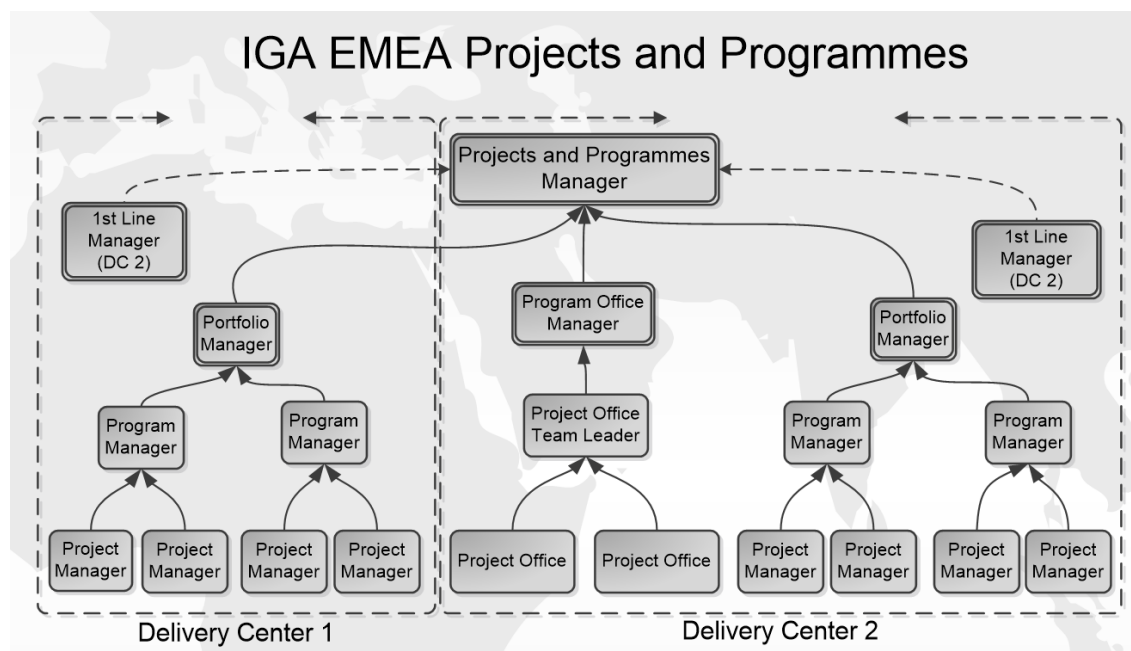
Organizačně IGA spadá pod obchodní skupinu IBM Global Technology Services. IGA se dále dělí podle lokací (IGA EMEA). Organizační jednotka nadřazená IGA EMEA Projects and Programmes se jmenuje IGA EMEA New Business (24).

2.1.4 Oddělení IGA EMEA Projects and Programs

Toto oddělení se zabývá řízením projektů, které mají za cíl standardizování a údržbu infrastruktury IBM, zajištění bezpečnosti a zvyšování efektivity a účinnosti. Zaměstnává přibližně 150 lidí v několika různých DC. Organizační struktura oddělení je, stejně jako v celém IBM, maticová. Oddělení dále můžeme dělit na portfolia, programy a Project Office.

Projects and Programmes Manager (PPM) je vedoucím celého oddělení. Manažeři první linie (1st Line Manager) se zodpovídají PPM, na základě pracovní smlouvy, především za personální a administrativní záležitosti. Jejich vlastní funkční manažer se ale nachází mimo zkoumané oddělení. Obvykle platí, že v jednom DC pracuje jeden manažer první linie z P&P. Všichni ostatní manažeři mají nejen svého funkčního manažera, ale také svého manažera první linie. Portfolio manažeři řídí svá portfolia - skupinu programů. Obdobně programoví manažeři řídí skupinu projektů. Projektoví manažeři již řídí jednotlivé, právě vykonávané projekty (24).

Program a Project Office jsou pozice, které nespádají pod žádné portfolio. Program Office Manager se zodpovídá přímo PPM. Project Office Team Leader poté spadá pod Program Office a řídí členy svého týmu. Mezi hlavní činnosti Program a Project Office patří například administrace databází, a různých systémů, využívaných tímto oddělením, podpora projektové administrace a podpora přidělování zdrojů (25).



Obrázek č. 5: Zjednodušené organizační schéma oddělení P&P (vlastní zpracování podle: 24, 25)

2.2 Popis řešeného problému

Na oddělení P&P vznikl nový nápad, kterým byl systém organizovaného hodnocení zaměstnanců oddělení, pro vnitřní potřeby oddělení s použitím vlastních otázek. Vzniklo několik formulářů, které si kladly za cíl, toto hodnocení získat. Mým úkolem byl návrh a tvorba systému pro hromadný sběr a zpracování těchto dat, který by pomohl zvýšit přehled vedení oddělení o zkušenostech zaměstnanců. Tento systém by měl využívat otázek z původních formulářů a mělo by se jednat o vlastní řešení.

2.2.1 Původní stav formulářů

Úplně prvním krokem, při tvorbě systému, bylo přesné zjištění původního stavu. Jelikož se jednalo o krátce používaný nápad, tak byl původní stav systému následující:

- Hodnocení probíhalo v menším, než požadovaném měřítku, například na úrovni programu.
- Několik podobných formulářů. Formuláře byly podobně členěné, a obsahovaly podobné otázky.
- Formuláře byly ve formátu xls. K vyplňování se používala aplikace Microsoft Excel nebo Apache Open Office Calc.
- Otázky se dělily do skupin. Příklad: úroveň anglického jazyka se dělí na mluvený a písemný projev daného manažera.
- Většina otázek byla hodnocena číselným hodnocením podle dané stupnice (0 - 8), několik otázek umožňovalo písemné hodnocení. Z většiny číselně hodnocených otázek se počítal průměr.
- Formuláře se shromažďovaly různým způsobem (email apod.).
- Minimální automatizace, případné analýzy by se musely provádět ručně.
- Hodnocení prováděné jednou za čtvrtletí.

Evaluation Scale: 0 - No skill - The employee has no academic or practical knowledge of the 1 - Acquired Beginner - The employee has gained elementary/limited knowh, training, mentoring, experiences or other skill development activities and has limited project. 2 - Acquired Experienced - The employee has gained satisfactory/acceptable education, training, mentoring, experiences or other skill development activities and hasent or project. 3 - Acquired Professional - The employee has gained comprehensive/compeducation, training, mentoring, experiences or other skill development activities and hasent or project. IMPORTANT: Wherever possible work with hard project evidence and data.	Project Manager Feedback											
	2014				2015				2016			
	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name	Name
English Language skills												
English Spoken - where 1 is Beginner and 10 Fully Fluent												
English Written - where 1 is Beginner and 10 Fully Fluent												
General Skills												
Communication, Negotiation and Assertiveness Skills												
Leadership Skills and Self Confidence												
Stamina, Pro-Activity and Reliability												
Smart and open minded (thinking out of the box), Innovations												
General IGA Awareness and Experience												
Years with IGA												
Opened Questions/Recommendations												
Overview of strengths and weaknesses												
Please provide comments to elaborate on your responses to the questions												
What suggestions for development do you have (e.g. what should the individual start, stop and/or continue doing)												
Total Average												

Obrázek č. 6: Ukázka původního formuláře (vlastní zpracování)

2.2.2 Požadavky na systém

Dalším krokem bylo stanovení základních požadavků na systém. Stanovil jsem následující.

- Jednoduchost: formuláře i program pro jejich zpracování musí být jednoduché a uživatelsky přívětivé.
- Jednotnost: vytvoření jednotného formuláře pro všechny zaměstnance.
- Automatizace: analýza formulářů i vyplňování hodnocení musí být co nejvíce automatizované.
- Přehlednost analýz: výsledky hodnocení by měly být poskytovány jako přehledná sada grafů. Písemné otázky by měly být podány v jednoduché sestavě.
- Bezpečnost: získávaná data jsou důvěrná, tudíž k nim nesmí mít přístup žádná neoprávněná osoba.
- Přenositelnost: k vyplňování vstupních dat by měl být použit běžně používaný program nebo služba, to samé platí i o výsledcích analýz. Instalace případného programu pro tvorbu analýz by měla být co nejjednodušší.
- Kapacita: formuláře a výstupy analýz by neměly být paměťově náročné.
- Rychlost: systém by měl rychle reagovat a analýza vstupních dat by měla být co nejrychlejší.
- Srozumitelnost: k celému systému by měla být vytvořena podrobná dokumentace.

Výstup systému

Výsledky hodnocení jsou určeny především PPM a manažerům první linie. Bylo by ovšem vhodné zpřístupnit výsledky také portfolio manažerům. Navrhovaný systém ale musí poskytovat manažerům pouze data relevantní k jejich pozici.

Po domluvě s vedením jsem stanovil následující okruhy grafů.

- Porovnání celkové úrovně DC, portfolií, programů.
- Porovnání průměrné celkové úrovně skupin dovedností a zkušeností.
- Vývoj průměrného hodnocení zaměstnance.
- Vývoj jednotlivých dovedností zaměstnance.
- Porovnání hodnocení poskytnutého manažery hodnotících stejného zaměstnance.

Požadavky na uložení

Rovněž si je třeba upřesnit požadavky na velikost uložení, neboť při výběru použitého řešení bude třeba vzít v potaz i předpokládaný počet záznamů.

Zkoumané oddělení zaměstnává asi 150 lidí. Hodnocení probíhá každé čtvrtletí. Počet otázek, které je třeba zodpovědět, se pohybuje kolem 30 otázek, z čehož 3 jsou hodnoceny písemně, a tudíž se musí uložit odděleně.

Po přidání 20% rezervy nám vychází 21600 záznamů za rok. Toto číslo bude v praxi jistě menší, ale umožňuje představu o požadavcích na kapacitu uložení.

2.2.3 Stanovení základních možností řešení

Důležitou částí vývoje bylo stanovení základních možností daného řešení, které umožní přesnější zaměření analýzy. Mezi možná řešení patří:

- Implementace do některého ze stávajícího systému.
- Vytvoření vlastní webové aplikace.
- Využití některého stávajícího síťového uložení a vytvoření desktopové aplikace pro zpracování takto shromážděných formulářů.

2.3 Analýza současného stavu hardwaru

2.3.1 Pracovní stanice

Na oddělení se používají různé notebooky značky Lenovo Thinkpad, například modely řady T430. Vzhledem k množství používaných a dostupných konfigurací budu počítat s následujícími parametry, které vycházejí z běžně dostupných konfigurací:

- Procesor: Intel Core i5-2520M (2,5 GHz, 3 MB L3).
- Operační systém: Windows 7 Professional (64-bit).
- Paměť: 8 GB DDR3.
- Úložení: 500 GB.
- Display: 14,0" HD+ (26).

Pokud k tomuto základu přidáme odolné šasi a dlouhou výdrž na baterii, tak dostáváme na kancelářskou práci více než dostačující notebook.

2.3.2 Počítačová síť a Servery

Pro připojení do intranetu a k internetu mohou zaměstnanci na pracovišti použít síť Wi-Fi nebo síťové kabely, u každého pracovního místa. Připojení do sítě je dimenzované pro rychlost přibližně 50 kb/s na zaměstnance a existuje proces pro zvýšení kapacity, pokud je spojení dlouhodobě vytížené. Dostupné je rovněž připojení přes VPN, pro práci mimo pracoviště (27).

Zkoumané oddělení nemá vlastní servery. Možnost tvorby serverových aplikací a jejich umístění na servery různých platform zde ale je, i když získání potřebných povolení je zpravidla delší proces (v řádu týdnů, až několika měsíců), daný velikostí firmy a důrazem na bezpečnost. K umístění aplikace na server je samozřejmě nutný řádný obchodní důvod. Dostupné testovací servery a servery používané pro běžný chod aplikací nabízí různé hardwarové konfigurace i operační systémy (24).

2.4 Analýza současného stavu softwaru

V následující kapitole popíši běžně dostupný Software, který by bylo možné pro daný systém použít.

2.4.1 Kancelářské balíky

Microsoft Office

Jedná se o placený kancelářský balík od firmy Microsoft. Zaměstnanci firmy o něj mohou požádat v různých verzích aktuálně dostupných verzích. Balík je dostupný ve dvou základních edicích: Standard a Professional, přičemž základním rozdílem jsou nabízené aplikace. Standardní edice nabízí následující aplikace:

- Word – textový procesor.
- Excel – tabulkový procesor.
- PowerPoint – umožňuje tvorbu prezentací.
- OneNote – aplikace pro tvorbu a ukládání poznámek.
- Outlook – groupware klient.
- Publisher – textový a grafický editor (28).

Profesionální edice navíc nabízí:

- Access – umožňuje správu relačních databází.
- InfoPath – nástroj pro tvorbu elektronických formulářů (28).

Nyní se zaměřím na potenciálně použitelné části tohoto kancelářského balíku.

Excel je tabulkový procesor. Umožňuje nejen využití nejrůznějších funkcí v tabulkách a jejich grafické úpravy, ale také tvorbu grafů, seznamů, souhrnů a kontingenčních tabulek a grafů.

Access je databázová aplikace. Umožňuje návrh, úpravu i výběr dat z relačních databází pomocí uživatelského rozhraní ale také SQL příkazů. Rovněž umí zobrazovat data pomocí grafů. Excel i Access umožňují vzájemnou výměnu dat (28).

Další vývoj aplikace InfoPath byl zastaven. Aplikace bude podporována do roku 2023, ale v dalších verzích balíku Microsoft Office se již vyskytovat nebude, tudíž její využití zvažovat nebudu (29).

Aplikace rovněž umožňuje využití jazyka Visual Basic for Application (VBA) pro tvorbu uživatelských maker a rozšíření funkčnosti aplikací. Dále rovněž umožňují propojení aplikací. Tvorba makra je v aplikaci aplikacích tohoto balíku možná dvěma základními způsoby. Nahráním a zápisem kódu. Režim nahrávání zaznamenává uživatelské akce v aplikaci a uloží je jako kód VBA. Zápis kódu probíhá ve vývojovém prostředí vestavěném do Excelu (4).

Microsoft Office 2013 navíc umožňuje používání cloudového úložiště Microsoft OneDrive.

Na rozdíl od jiných kancelářských balíků je Microsoft Office dostupný pouze pro operační systémy Windows (ve zvláštní verzi také pro OS X), i když již existuje webová i mobilní verze systému. Ty ovšem neumožňují využití maker (28).

Apache OpenOffice

Jedná o volně šiřitelný kancelářský balík poskytovaný The Apache Software Foundation. Jedná se asi o největší konkurenci balíku Microsoft Office, protože balík je zdarma, dostupný na většinu operačních systémů a nabízí podobné možnosti i aplikace.

- Writer – textový procesor.
- Calc – tabulkový procesor.
- Impress – umožňuje tvorbu prezentací.
- Draw – aplikace pro tvorbu grafiky.
- Base – umožňuje práci s databázemi.
- Math – aplikace pro zápis matematických rovnic (30).

Využitelnými aplikacemi balíku by mohly být Calc a Base. Nabízejí podobné možnosti jako Excel a Access. Base na rozdíl od Accessu používá jazyka HSQLDB. OpenOffice rovněž umožňuje tvorbu uživatelských maker, podobně jako Microsoft Office, ale ve vlastní verzi jazyka Basic (30).

Srovnání Microsoft Office a Apache OpenOffice

Oba tabulkové procesory nabízí téměř stejnou kapacitu řádků na pracovní list, tj. asi 1,4 miliónů řádků. Access je omezen velikostí celé databáze na 2 GB, Base není pevně omezen. Nyní porovnáme silné a slabé stránky obou balíků, přičemž budu vycházet z vlastních zkušeností.

Tabulka č. 1: Srovnání balíků Microsoft Office a Appache OpenOffice

	Microsoft Office	Apache OpenOffice
Silné stránky	Rychlé zpracování dat. Přívětivé uživatelské prostředí. Placená podpora.	Dostupný zdarma. Dostupnost pro více platforem.
Slabé stránky	Placený. Pouze pro Windows a OS X.	Pomalejší odezva. Menší možnosti. Menší uživatelská přívětivost.

(Vlastní zpracování)

Microsoft Office téměř všichni zaměstnanci mají a používají, a tudíž omezení v podobě placených licencí nemusím brát v potaz. Na druhou stranu často jedná o standartní edice, které neobsahují Access, a tudíž by mohlo být nejvhodnější použít aplikaci Excel.

2.4.2 IBM Connections

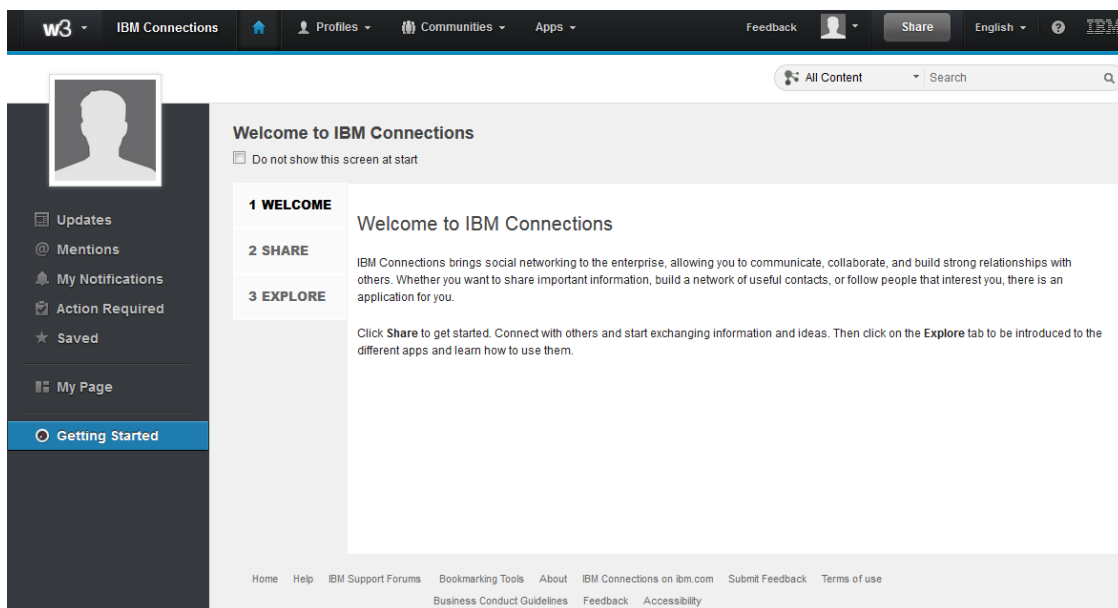
Jedná se o podnikovou sociální síť, vytvořenou společností IBM a dostupnou všem jejím zaměstnancům. Síť zaměstnancům umožňuje komunikaci, sdílení souborů, kooperaci na projektech a mnoho dalšího. Síť umožňuje přístup přes webové rozhraní i mobilní zařízení. V současné době je aktuální verze 5, přičemž se zaměříme na aspekty sítě, které bych mohl použít pro navrhovaný systém (20).

Síť používá systém komponent, který umožňuje uživatelům snadnou implementaci potřebné funkcionality na jejich stránky či tzv. komunity (virtuální skupiny uživatelů). Následuje přehled nejdůležitějších komponent:

- Home page – domácí stránka, ukazuje základní data důležitá pro uživatele.
- Profiles – umožňuje procházení profilů uživatelů.
- Communities – umožňují tvorbu a správu komunit.
- Activities – komponenta pro správu a přidělování úkolů.
- Files – cloudové úložiště souborů. Umožňuje ukládání soukromých a veřejných souborů a také jejich sdílení s ostatními uživateli. Lze využít i pro důvěrná data.
- Docs – obsahují textový i tabulkový procesor i nástroj pro tvorbu prezentací. Obsahuje základní funkce a nenabízí možnost uživatelských maker (20).

IBM Connections nabízí i řadu dalších komponent, ale toto jsou ty nejdůležitější. Sociální síť je stále ve vývoji a přibývá do ní nová funkcionality (20).

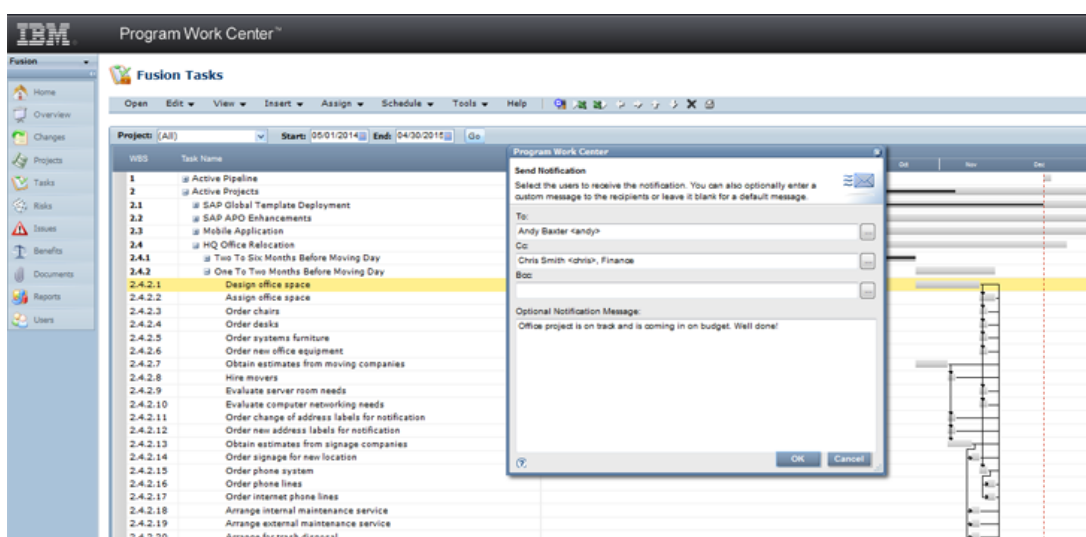
Pro zadaný systém se jeví jako nejvhodnější využití komponenty Files pro ukládání souborů ve spojení s komponentami Communities a Activities pro snazší přidělování tohoto úkolu zaměstnancům.



Obrázek č. 7: Ukázka IBM Connections (20)

2.4.3 IBM Program Work Center

IBM Program Work Center (IPWC) je webová služba vytvořená a nabízená společností IBM ve formě SaaS. IPWC je nástroj pro kontrolu programů i jejich projektů. Nabízí možnosti zaznamenávání, sledování i schvalování milníků, rizik projektů apod. mezi členy týmu v reálném čase. Také nabízí souhrnný pohled na programy i portfolia a export záznamů do aplikace Microsoft Excelu, Project nebo do formátu PDF. IPWC je stále ve vývoji a tudíž dochází k přidávání nové funkcionality a vylepšování té stávající (20). Tato služba je rovněž používána na oddělení P&P všemi zaměstnanci.



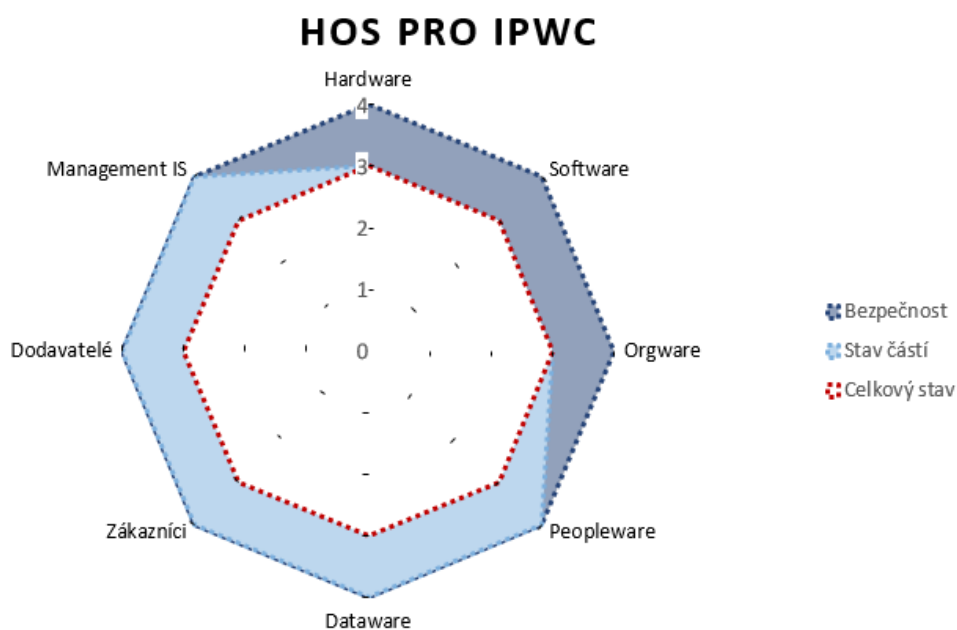
Obrázek č. 8: Ukázka uživatelského prostředí IPWC (20)

IPWC umožňuje sledovat položky projektů, kterými jsou například úkoly, rizika, problémy, benefity, dokumenty nebo hlášení.

HOS 8 analýza systému IPWC

V následující části této kapitoly popíši HOS 8 analýzu této aplikace, kterou jsem zpracoval pomocí portálu Zefis (18). Ta mi umožnila lépe pochopit současný stav systému a zjistit, zda by bylo vhodné hodnocení zaměstnanců zabudovat do tohoto systému. Otázky zodpověděl Michal Bratko (31), jeden ze zaměstnanců oddělení.

Z grafu můžeme vidět, jak dopadla analýza systému. Stav Hardwaru, Softwaru a Orgwaru je spíše dobrý, ostatní části systému dopadly dobře. Bezpečnost systému je rovněž podle této analýzy dobrá. Vzhledem k tomu, že celkový stav se hodnotí podle nejslabší části, tak stav systému hodnotím jako spíše dobrý. Pokud ale vezmeme v úvahu, že systém je stále ve vývoji a dochází k přidávání další i vylepšování již stávající funkcionality, tak to považuji za velmi dobrý stav.



Graf č. 1: Výsledek HOS 8 analýzy IPWC (vlastní zpracování podle: 18)

Jak již bylo řečeno, IPWC je stále ve vývoji a tudíž by bylo možné navrhnout zabudování systému hodnocení zaměstnanců přímo do něj. V tomto případě by bylo nutné navrhnout novou část této webové služby, která by umožňovala zadávání hodnocení a i zpracování výstupů.

2.4.4 IBM Bluemix

IBM Bluemix je cloudová platforma společnosti IBM pro hostování webových a mobilních aplikací, která v sobě integruje také DevOps. Platforma v sobě integruje řadu nástrojů pro snazší, rychlejší a levnější vývoj aplikací a jejich hostování, v řadě různých programovacích jazyků a s pomocí nejrůznějších služeb. Spuštěným aplikacím lze měnit velikost přidělené paměti a další parametry podle aktuální potřeby. Celá platforma je postavená na cloudové infrastruktuře společnosti SoftLayer (dceřiná společnost společnosti IBM) a umožňuje tvorbu veřejných i soukromých projektů. Bluemix je zákazníkům poskytován jako PaaS, DevOps jako SaaS, přičemž se platí za objem používaných služeb (množství použité paměti v čase, počty uživatelů atd.). Technologie poskytované uživatelům platformy se dají rozdělit na technologie poskytované přímo organizací IBM a technologie třetích stran. Následující přehled obsahuje některé vybrané technologie integrované v IBM Bluemix (32).

- Běhová prostředí (Runtimes).
 - Node.js – framework pro vývoj síťových aplikací.
 - PHP – skriptovací jazyk pro webové aplikace.
- Watson – umožňuje tvorbu kognitivních aplikací pomocí systému IBM Watson.
- Webové aplikace – služby pro podporu webových aplikací.
 - Data Cache – urychluje aplikace ukládáním do rychlejší paměti platformy.
- Data Management.
 - SQL database – relační databáze založené na řešení DB2.
 - ClearDB MySQL database – databáze postavená na MySQL.
 - Cloudant NoSQL DB – databáze postavená na principu NoSQL.
- Big Data – technologie pro zpracování „velkých dat“.
 - DashDB – relační databáze umožňující analýzy uložených dat.
- Business Analytics – nástroje pro analyzování dat z databází.
 - Embeddable Reporting – analýzy dat pomocí systému IBM Cognos Business Intelligence.
- Bezpečnost.
 - Single Sign On – služba pro autentizaci uživatelů. Umožňuje napojení na vlastní nebo externí autentizační zdroje, např.: LinkedIn (32).

Princip vývoje v IBM Bluemix

IBM Bluemix není přímo vývojová platforma, ale slouží pro hostování aplikací zákazníka. Samotný vývoj aplikace se tudíž odehrává v samostatném vývojovém prostředí a aplikace jsou následně do platformy nahrány. Pro vývoj aplikací je potřeba internetový prohlížeč a příkazový řádek pro Cloud Foundry (Cloud Foundry je open source cloudová platforma, na které je platforma IBM Bluemix postavená).

Uživatel služby tudíž vytvoří kód pro svoji aplikaci ve zvoleném jazyce a vývojovém prostředí. Po přihlášení do webového rozhraní platformy může pro tuto aplikaci zaregistrovat doménu, alokovat potřebné služby a následně již aplikaci nahrát. Po rozběhnutí aplikace je ji možné i nadále upravovat, přidělit jí více paměti nebo třeba další služby (32).

Zpoplatnění služby

Při zpoplatnění mnoha služeb na IBM Bluemix se používají tzv. GB-hodiny. Ty se počítají jako doba, kterou aplikace běžela (v hodinách), krát velikost paměti, kterou aplikace potřebovala (v GB), krát počet instancí aplikace. Další služby jsou zpoplatněny velikostí potřebné paměti (zpravidla databáze), počtem uživatelů (bezpečnost), nebo požadavků na danou službu (analytické služby).

Všechny služby také poskytují určitou úroveň užívání, za kterou uživatel neplatí. Pro běhová prostředí se tato úroveň v současné době pohybuje na 375 GB-hodinách za měsíc (32).

2.4.5 Vybrané technologie pro vývoj webových aplikací

PHP

Hypertext Preprocessor, neboli PHP, je skriptovací jazyk. Nejčastěji je využíván při vývoji dynamických webových stránek. Používá se v kombinaci s HTML. PHP zajistí vygenerování (získání) potřebných dat, například z připojené databáze (například MySQL databáze) a na straně serveru doplní HTML kód o proměnlivé údaje. Server poté hotovou webovou stránku pošle klientovi. Pro navrhovaný systém bych zvolil verzi 5.6, která je nejnovější, a její podpora bude trvat nejdéle (33).

Node.js

Node.js je událostní framework postavený na běhovém prostředí JavaScriptu, který se pro tvorbu webových aplikací, běžících na straně serveru. Při programování aplikací se tedy používá jazyk JavaScript, což je objektově orientovaný skriptovací jazyk, často používaný pro tvorbu dynamických webových stránek (34).

Srovnání PHP a Node.js

PHP se na webu používá přibližně 20 let, zatímco Node.js je používán pouze pár let. Vlastnosti obou jazyků se pokusím nastínit následující tabulkou.

Tabulka č. 2: Porovnání vlastností PHP a Node.js

PHP	Node.js
Jednodušší, více doplňků.	Modernější.
Snazší přístup k databázím.	Možnost používat JSON (JavaScript Object Notation).
Rychlejší vývoj.	Rychlejší běh aplikace.

(Vlastní zpracování podle: 35)

MySQL

MySQL je open source databázový systém, který je v některých verzích dostupný zdarma (MySQL Community Edition) a velmi často se používá při tvorbě webových stránek ve spojení s PHP. MySQL umožňuje tvorbu relačních databází s použitím jazyka SQL (36).

2.5 Souhrn analýzy

Tuto práci tvořím pro oddělení IGA EMEA Projects and Programmes společnosti IBM. Společnost jako taková se zabývá prodejem vlastních výrobků a služeb z oblasti informačních technologií. Zkoumané oddělení patří do účtu IGA, který poskytuje služby a vybavení ostatním částem IBM, a oddělení samotné řídí projekty, které mají již zmíněné na starosti.

Oddělení vede Projects and Programme manager. Manažeři první linie, v každé lokalitě, se mu zodpovídají a mají na starosti personální záležitosti zaměstnanců. Zaměstnanci oddělení se dále dělí do portfolií, která účelově slučují programy a ty slučují projekty. Bokem této struktury stojí Programmes and Project Office.

Systém hodnocení zaměstnanců, který budu tvořit, bude založen na původních dotaznících používaných pro tento účel. Rovněž jsem si stanovil podmínky, které musím na systém klást, jako například uživatelskou přívětivost nebo bezpečnost získaných dat. Data budou sbírána čtvrtletně a výsledky musí být dostupné vedení oddělení (PPM a manažerům první linie).

V předposlední části této kapitoly jsem se zaměřil na současný stav systémů používaných na zkoumaném oddělení. Stav používaných notebooků shledávám jako plně dostačující, rovněž existuje možnost tvorby webových aplikací. Rovněž jsem porovnal běžně používané kancelářské balíky, popsal podnikovou sociální síť (IBM Connections) a webovou službu IPWC, u které jsem rovněž provedl analýzu HOS 8. Poté jsem popsal platformu IBM Bluemix a technologie, které by šly využít pro tvorbu webové aplikace.

Z daného mi vycházejí následující možnosti tvorby aplikace.

- Implementace do IPWC nebo podobného systému.
- Samostatná webová aplikace s použitím PHP a MySQL.
- Webová aplikace.
 - Aplikace hostovaná na běžném serveru.
 - Aplikace běžící na platformě IBM Bluemix.
- Desktopová aplikace naprogramovaná do aplikace MS Excel nebo MS Access, v kombinaci s ukládáním dat do IBM Connections.

Tato řešení podrobněji popíši v následující kapitole, a poté se zaměřím na vybrané řešení.

3 VLASTNÍ NÁVRH ŘEŠENÍ, PŘÍNOS PRÁCE

Jak již napovídá název této kapitoly, na následujících stránkách se budu zabývat výběrem a návrhem daného systému. Před samotným ekonomickým zhodnocením ještě stručně nastíním řešení, které by bylo možné použít v budoucnu.

3.1 Možnosti řešení

Již v průběhu předchozí kapitoly jsem zmínil několik řešení, která se mi pro tvorbu systému pro hodnocení zaměstnanců zdají jako nejvhodnější. Nyní se na tato řešení podíváme podrobněji.

3.1.1 Implementace do IPWC

První možné řešení je implementace systému do IBM Program Work Center. Jedná se o nástroj pro správu portfolií, programů, projektů a ukládání příslušné dokumentace. Hodnocení zaměstnanců, kteří pracují nebo pracovali na pro dané oddělení, se mi tedy jeví jako rozšíření stávající funkcionality než jako změna zaměření tohoto systému. Díky tomu, že je IPWC stále ve vývoji, je navíc možné žádat o přidání nové funkcionality. Na druhou stranu je nutné podotknout, že IPWC není personálním systémem, a tudíž by přidání navrhované funkcionality nemuselo být vývojáři přijato.

Pro přidání funkcionality pro hodnocení zaměstnanců by bylo nutné přidání dalších tabulek do databáze systému (tj. tabulky pro ukládání otázek a hodnocení). Poté by bylo potřeba přidat tuto funkcionalitu do uživatelského rozhraní. To by nyní obsahovalo dvě nové položky, kterými by byly „Vytvoření nového hodnocení“ a „Prohlídka výsledků“, u které je potřeba dbát na zobrazování výsledků pouze příslušným pracovníkům. Další potřebnou funkcionalitou by bylo nastavení automatických žádostí o zadání nového hodnocení.

Mezi největší výhody tohoto řešení patří:

- Rychlejší implementace, protože některé části systému jsou již hotové.
- Uživatelsky příjemné řešení, protože se jedná o již známý systém.
- Není nutné řešit přenositelnost (systém používá webové rozhraní).
- Aplikace by mohla být přístupná pro další oddělení a zákazníky společnosti.

Nevýhody řešení:

- Nejistý vývoj aplikace.
- Vývoj by probíhal mimo zkoumané oddělení.

3.1.2 Webová aplikace

Další možností je tvorba nové webové aplikace. Toto řešení mohu dále rozdělit na dvě samostatná řešení:

- Řešení umístěné na „běžném“ serveru.
- Řešení umístěné na platformě IBM Bluemix.

Aplikace běžící na běžném serveru

Tímto řešením chápeme umístění aplikace na fyzický nebo virtuální server. Hlavní nevýhodou tohoto řešení může být delší proces umístění aplikace na server. Při vývoji aplikace by také bylo nutné zajistit zabezpečení přístupu k aplikaci, pravděpodobně za využití stávajících autentizačních nástrojů společnosti.

Aplikace umístěná na platformě IBM Bluemix

IBM Bluemix funguje na principu PaaS. Úkolem vývojáře a administrátora by tudíž byla jen údržba aplikace. Hlavní výhodou tohoto řešení je možnost výběru z mnoha modulů, které je možné použít při vývoji na této platformě. Potenciální nevýhodou řešení ale může být fakt, že se jedná o novou platformu.

Dalším důležitým krokem při návrhu systému (obou řešení) by bylo vytvoření databázové struktury. Hlavními tabulkami databáze by byly tabulky zaměstnanců, seznamu otázek a tabulka vyplněného hodnocení. Tabulku zaměstnanců by rovněž bylo vhodné napojit na interní databázi zaměstnanců. V opačném případě by administrátor musel udržovat seznam zaměstnanců aktuální.

Výhody serverové aplikace:

- Časově méně náročné pro administrátora.
- Možnost prodeje řešení nebo rozšiřování možností aplikace.
- Snadná přenositelnost výsledků.

Nevýhody těchto řešení:

- Potřeba napojení aplikace na interní databáze společnosti.
- Složitější implementace zabezpečení (v případě nahrání na „běžný“ server).
- IBM Bluemix je nová platforma.

3.1.3 Systém založený na aplikaci Microsoft Excel a IBM Connections

Posledním možným řešením, které uvedu před výběrem vlastního řešení, je desktopová aplikace a využití cloudového uložiště pro ukládání výsledků hodnocení. Jako nejvhodnější aplikací pro tvorbu desktopové aplikace se mi jeví aplikace Microsoft Excel. Excel se mi jeví vhodnější než Access, a to proto, že Excel je pro zaměstnance dostupnější než Access, a rovněž je jim jeho uživatelské prostředí více známé. Výhoda databázového řešení se mi zde nezdá být tak velká, spíše naopak, Access by mohl návrh celé aplikace ztížit a zároveň snížit přenositelnost výsledků.

Systém by v tomto případě bylo nutné rozdělit do několika částí. Do formuláře by zaměstnanci vyplňovali jejich hodnocení. Ten by poté ukládali do příslušné komunity v IBM Connections. Další částí by byl program pro zpracování výsledků hodnocení. To by mělo za úkol načtení dat z dotazníků a zpracování těchto dat do požadovaných grafů a tabulek.

Výhody:

- Využití stávajícího softwaru oddělení.
- Rychlá implementace (v řádu týdnů).
- Snadná přenositelnost programu pro zpracování (makra v Excelu není potřeba instalovat).
- Vychází z původních dotazníků.
- Možnost snadného používání kontingenčních tabulek.

Nevýhody:

- Oproti předchozím řešením nižší stupeň automatizace (pro uživatele i administrátora).
- Aplikace bude pracovat pouze na systémech Windows a OS X.

3.1.4 Výběr řešení

Po zvážení všech popsaných řešení jsem se s vedením oddělení dohodl na následujícím.

Pro mnoho firem by se jako nejvhodnější řešení mohlo jevit umístění aplikace na vlastní server. Pro oddělení P&P jsem toto řešení nedoporučil, protože umístění aplikace na běžný server by mohlo trvat delší dobu a řešení postavené na platformě Bluemix by mohlo být vhodnějším řešením.

Vhodným řešením by rovněž mohla být implementace systému do IPWC. Systém je již pracovníkům oddělení známý, je již používán a je do něj stále možné přidávat další prvky. Na druhou stranu, přidání navrhované funkcionality není jisté a vývoj by se odehrával mimo oddělení.

Asi nejvhodnějším řešením by mělo být řešení postavené na platformě IBM Bluemix. Oproti IPWC by tvorba systému mohla být plně v režii oddělení. Toto řešení navíc umožňuje rozšiřování aplikace přesně podle potřeb oddělení.

U všech řešení ale zatím neznáme jejich reálný přínos a jejich vývoj i údržba se prozatím jeví jako zbytečně nákladné a zdlouhavé. Proto byla zvolena poslední možnost, kterou je kombinace aplikace postavené na Microsoft Excel a sociální síť IBM Connections. Toto řešení je možné implementovat nejrychleji (v řádu týdnů) a s nejmenšími náklady. Nejedná se sice o nejvhodnější dostupné řešení, ale umožní stanovení reálných přínosů řešení pro dané oddělení a sběr dat do doby vytvoření nového systému. Data získaná tímto systémem jsou navíc snadno implementovatelná do případné nové verze systému.

V případě, že by se tento systém osvědčil, bych navrhoval stavbu vlastní webové aplikace umístěné na platformě IBM Bluemix. Tomuto řešení se budu krátce věnovat v závěru kapitoly. Před samotnou tvorbou takového řešení by ale bylo vhodné provést analýzu procesů, probíhajících na oddělení, a zjistit, zda zde existují další procesy vhodné pro zabudování do nového systému.

3.2 Popis vybraného řešení

Nyní se zaměřím na popis vybraného řešení. Začnu nástinem systému a stručným popisem hlavních komponent.

Uživatele systému můžeme rozdělit do 3 základních skupin:

- Hodnotitelé - vyplňují hodnocení zaměstnanců. Hodnotícími jsou manažeři programů a portfolií, manažer Program Office a PPM.
- Vlastníci - jsou hlavními příjemci výstupů hodnocení. Jedná se o vedení oddělení.
- Administrátor - má na starosti sběr a zpracování dat a aktualizace dat v systému.

Navrhovaný systém se skládá z následujících částí:

- Formulář - soubor aplikace Excel, pro vyplňování zaměstnaneckého hodnocení.
- Program na zpracování hodnocení - program napsaný v jazyce VBA aplikace Excel, které provede zpracování dat. Dále jen program Zpracování.
- Komunita v IBM Connections - bude sloužit jako uložisko pro všechna data systému.
- Orgware - poslední částí systému bude podrobná dokumentace. Ta vyplývá z tohoto návrhu, a proto se jí v této práci nebudu dále zabývat.

3.2.1 Postup provádění hodnocení a analýzy

První krok, který administrátor musí provést, je rozeslání žádosti o vyplnění hodnocení hodnotitelům. Ti si z Komunity stáhnou svůj Formulář, vyplní požadované údaje a Formulář opět uloží do Komunity. Administrátor, po obdržení všech formulářů, stáhne aktuální verze formulářů.

Program Zpracování bude nejprve potřeba nastavit, a poté spustit nad aktuálními Formuláři. Program vytvoří výstupní soubory, které administrátor rozešle příslušným vlastníkům.

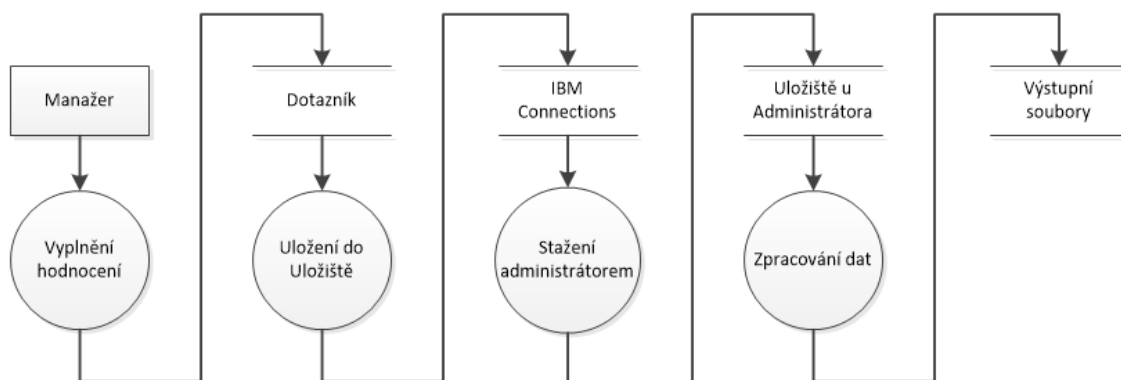


Diagram č. 1: Diagram toku dat v navrženém systému (vlastní zpracování)

3.2.2 Stanovení výstupních atributů hodnocení

Prvním krokem bylo stanovení dat, která systém bude získávat, abych později mohl upravit formuláře a vytvořit vhodnou strukturu datového uložení. Také bylo třeba určit datový typ dat, protože podle něj se budu omezovat vstupní hodnoty.

Tabulka č. 3: Atributy, potřebné pro provedení hodnocení

Atribut	Popis	Datový typ
Hodnotitel	Jméno hodnotícího manažera.	Řetězec
Hodnocený	Jméno hodnoceného manažera.	Řetězec
Úroveň hodnocených	Úroveň hodnocených manažerů.	Řetězec
DC	Delivery Center, ve kterém hodnocený pracuje.	Řetězec
Portfolio	Portfolio, pro které hodnocený manažer pracuje.	Řetězec
Program	Program, pro který manažer pracuje (platí pouze pro projektové manažery).	Řetězec
Rok	Rok, za který bylo hodnocení provedeno.	Číslo
Čtvrtletí	Čtvrtletí, za které byl hodnocení provedeno.	Řetězec
Okruh otázek	Název skupiny otázek. Definiuje obecnější skupinu znalostí/dovedností manažera.	Řetězec
Otázka	Rozvedení předchozího. Definiuje samotné otázky.	Řetězec
Hodnocení	Číselné nebo textové hodnocení dané otázky.	Řetězec / číslo (0-8)

(Vlastní zpracování)

3.3 Návrh formuláře

Při návrhu nového formuláře jsem vycházel z původních formulářů. Z předchozí kapitoly vyplývá, jaké údaje manažeři budou vyplňovat. V původních dotaznících nejsou explicitně uvedené následující údaje:

- Hodnotitel – jméno hodnotitele bylo obvykle obsaženo v názvu souboru.
- Úroveň hodnocených – vzhledem k původnímu rozsahu použití nebylo třeba.
- Delivery Center.
- Portfolio.
- Program.

Pro systém jsem se rozhodl sjednotit původní seznamy otázek (a nechat i původní členění na textové a číselné otázky) a přidat chybějící parametry. Protože některé parametry jsou pro celé hodnocení společné (Hodnotitel, Úroveň hodnocených, Portfolio), tak je tyto údaje lepší zadat pouze jednou, a to na úvodním listu sešitu formuláře (dále: „Obecné Informace“). Těmto údajům budu dále říkat „Údaje sešitu“. Zde jsou také uložena některá další data, potřebná pro fungování dotazníku, stejně jako tlačítka pro spuštění některých podprogramů formuláře.

Druhým typem listu jsou listy pro zapsání hodnocení – list „Hodnocení“. Tento list je založen na původních formulářích s několika úpravami. Do listu jsem přidal oblast pro zadání DC, programu (pokud je třeba) a jména hodnoceného. Této trojici argumentů budu pro dále říkat „Údaje hodnoceného“. Jeden list bude schovaný a bude sloužit jako předloha, ze které se pro každého dalšího hodnoceného manažera vytvoří kopie.

Do Formuláře jsem přidal různé procedury, které zajišťují jednodušší vyplňování. První z nich přidává další listy pro hodnocení. Další metoda by měla provést základní kontrolu vložených dat.

A		B		C	D	E	F	G	H	I	J
Evaluation Scale: 0 - No skill 1 - Acquired Beginner 2 - Acquired Experienced 3 - Acquired Professional 4 - Applied Beginner 5 - Applied Experienced 6 - Applied Professional 7 - Mastered 8 - Innovative Leadership Exception from this scale: Years with IGA (see its comment). IMPORTANT: Wherever possible work with hard project evidence and data.				Delivery centre:		Select					
						Select					
				Evaluated's Name:		Select					
				PM Feedback for: Not specified							
				2015				2016			
				Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
				Not evaluated	Not evaluated	Not evaluated	Not evaluated	Not evaluated	Not evaluated	Not evaluated	Not evaluated
English Language skills											
English Spoken											
English Written											
General IGA Awareness and Experience											
Years with IGA											
Opened Questions/Recommendations											
Overview of strengths and weaknesses											
Please provide comments to elaborate on your responses to the questions above											
What suggestions for development do you have											
Total Average				No data	No data	No data	No data	No data	No data	No data	No data

Obrázek č. 9: Ukázka listu Hodnocení (vlastní zpracování)

3.3.1 Ošetření vstupních hodnot

Zásadním problémem tohoto systému je zajištění konzistentních vstupních dat. Jako asi nejpravděpodobnější zdroj chyb se jeví zadání textového hodnocení do pole pro číselné hodnocení a zadání hodnocení mimo rozsah stupnice. Dalším problémem jsou názvy a jména. Například DC mají různé názvy a formy zápisu. To samé platí i pro jména manažerů.

Řešení prvního problému je následující. Buňky v listech hodnocení jsem nastavil pro kontrolu vstupních hodnot. Pro číselné hodnocení jsem nastavil kontrolu vstupních hodnot na celá čísla od 0 do 8. Hodnoty textového hodnocení nelze nijak omezit, protože nelze stanovit, jaká vstupní hodnota je správná.

S tímto souvisí neoprávněné změny seznamu otázek. Zde stačil zákaz změny daných buněk (například pomocí nastavení ověření dat na náhodný řetězec).

Zbylé atributy bylo nutné omezit rozsahem možných hodnot. Některé hodnoty se ale mohou často měnit. Tudíž hodnoty, které se téměř nikdy nemění, jsou do formuláře zadány přímo (jedná se jména portfolio a úrovně hodnocených), a zbylé se musí vždy při spuštění souboru aktualizovat. Potřebné hodnoty se zadávají pomocí „drop-down listů“.

Aktualizace Formuláře

Aktualizace vstupních dat se provádí při otevření Formuláře. Formulář se uživatele zeptá na jeho přihlašovací jméno a heslo do IBM Connections s pomocí čehož se následně přihlásí do systému a stáhne soubor se proměnlivými údaji. Pomocí těchto údajů poté provede aktualizaci dat ve formuláři. Tento soubor bude obsahovat aktuální seznam manažerů pracujících pro oddělení, seznam DC a seznam programů.

Sjednocení dotazníku

Některé otázky jsou relevantní pouze pro některé manažery. Na základě vstupních dat je tudíž tyto otázky potřeba skrýt, aby je nevyplnili manažeři, kterých se netýkají. To provede metoda reagující na změny v listu obsahujícím obecné informace. Tento list bude rovněž obsahovat příslušné seznamy (seznam portfolií, a úrovní hodnocených), které budou rovněž zajišťovat správné zobrazení formuláře, pomocí intervalů řádků, které se mají skrýt nebo zobrazit.

3.3.2 Dodatečné úpravy formuláře

Na listech Hodnocení jsem do sloupce A vložil druh vkládaných dat. Tato data musí být rovněž nastavena tak, aby nešla změnit a skryta (zde bude stačit bílé písmo na bílém podkladu a zakázání změn). Bližším vysvětlením významu tohoto sloupce se budu zabývat později.

		Q1	Q2	Q3
		No Name	Not evaluated	Not evaluated
	Choose TRUE for every quarter for which you have written your evaluation.	PRAVDA	NEPRAVDA	NEPRAVDA
A	English Language skills			
S	English Spoken	7		
S	English Written	6		

Obrázek č. 10: Úpravy provedené na listu Hodnocení (vlastní zpracování)

Do Hodnocení jsem také přidal řádek, označující, zdali byl zaměstnanec v daném čtvrtletí hodnocen. To povede ke zrychlení zpracování dat. Pro tento řádek jsem navrhl funkci, která zjišťuje, jestli sloupec obsahuje nějaká data.

**= KDYŽ(NEBO(SUMA(C\$11:C\$125) > 0; NE(JE.PRÁZDNÉ(C\$126));
NE(JE.PRÁZDNÉ(C\$127)); NE(JE.PRÁZDNÉ(C\$128))); PRAVDA; NEPRAVDA)**

Kód č. 1: Funkce pro kontrolu obsahu dat (vlastní zpracování)

První část výrazu stanoví, jestli bylo zadáno číselné hodnocení. To zjistím tak, že suma hodnot v oblasti musí být větší jak nula (což je implicitní hodnota buňky pro funkci SUMA v Excelu). Poté kontroluji, jestli jsou pole pro textové hodnocení vyplněná, nebo ne. Vše je dohromady sloučené pomocí operátorů NEBO a NE (negace).

Další přidané podprogramy

Sešit obsahuje také několik dalších skriptů (a příslušná tlačítka). Jeden zajišťuje přidání nového listu pro hodnocení. Při stisknutí tlačítka na listu s obecnými údaji, se zobrazí dialogové okno se seznamem jmen. Po vybrání jednoho jména skript okopíruje vzorový dotazník a doplní do něj vybrané jméno.

Další tlačítko by mělo spustit skript pro základní kontrolu sešitu. Skript projde všechny listy a zkontroluje, jestli obsahují potřebné údaje. Sešit by měl dále obsahovat tlačítka a skripty pro manuální aktualizaci dat. Jedno z tlačítek tedy pouze spustí již popsany skript pro aktualizaci dat (pro případ, že by aktualizace neproběhla automaticky) a druhé umožní nahrání již staženého seznamu. Poslední tlačítko umožní nastavení uložení seznamu.

3.4 Návrh aplikace na zpracování formulářů

Program Zpracování, který zpracovává data, se skládá ze dvou hlavních částí. Jako první se provede načtení dat do aplikace Excel (Import), kdy se prochází a ukládají všechna data uložená ve formulářích. Druhou částí poté je vytvoření souborů s analýzou (Export). Při exportu se data dělí podle požadovaných atributů, aby se zachovala důvěrnost dat. Vybraná data se poté kopírují do výstupních souborů, a poté se vytvoří žádané grafy. Před spuštěním makra a po jeho ukončení se ještě musí provést některé pomocné podprogramy.



Diagram č. 2: Tok dat v programu Zpracování (vlastní zpracování)

Samotný sešit, ve kterém je program uložen, se skládá z několika listů. První list obsahuje ovládací prvky, proměnné nastavení (skryté) a výpis běhu programu. Další listy ukládají data získaná během importu. Listy pro ukládání dat mají strukturu shodnou s tabulkou atributů. Excel totiž neumožňuje SQL dotazy a jednoduchost řešení zde převáží nad ušetřeným místem. Následuje seznam všech listů programu Zpracování.

- Menu – slouží k ovládání programu a uložení nastavení.
- Průměrovaná data – uložisko pro číselně hodnocené otázky, ze kterých se počítá celkový průměr.
- Neprůměrovaná data – uložisko zbylých číselně hodnocených otázek.
- Textová data – uložisko odpovědí na otevřené otázky.

3.4.1 Podpůrné podprogramy

V průběhu návrhu programu jsem narazil na potřebu vytvoření různých podpůrných mechanismů. Nejdůležitější z nich nyní popíši.

Prvním krokem po spuštění programu je načtení nastavení. Uložisko nastavení se nachází na listu Menu a je před uživatelem skryto. Uživatel jej může změnit ve zvláštním formuláři. Hned po zjištění nastavení se uživateli zobrazí výpis nastavení a dotaz, zda může program pokračovat. Poté se provede několik řádků kódu pro zrychlení běhu a ošetření nežádoucích událostí. Na konci běhu všechny tyto hodnoty vrátím do původního stavu, nastavením na výchozí (opačné) hodnoty.

Application.ScreenUpdating = False

Application.Calculation = xlCalculationManual

Application.EnableEvents = False

Application.DisplayAlerts = False

Application.AskToUpdateLinks = False

Kód č. 2: Nastavení Excelu po spuštění Zpracování (vlastní zpracování podle: 3)

První řádek programu zrychluje běh makra tím, že aplikaci Excel zakáže vykreslování aktuálního stavu aplikace. Druhý řádek zakazuje automatické přepočítávání hodnot v otevřených sešitech. Třetí řádek zakazuje reakci kódu na události aplikace Excel, například na otevření sešitu. Poslední dva řádky poté zakáží zobrazování výstražných hlášení a dotaz pro aktualizaci odkazů v právě otevíraném sešitu.

Další možné procedury a moduly mají různé účely a podporují jak export, tak import. Jedná se například o proceduru kontrolující existence listu v zadaném sešitu, nebo metodu pro nalezení prvního prázdného řádku v tabulce. Také je potřeba vytvořit zvláštní modul pro konstanty. Odvoláváním se na konstanty se ulehčí práci v případě potřeby změny názvu některého prvku programu nebo jeho nastavení.

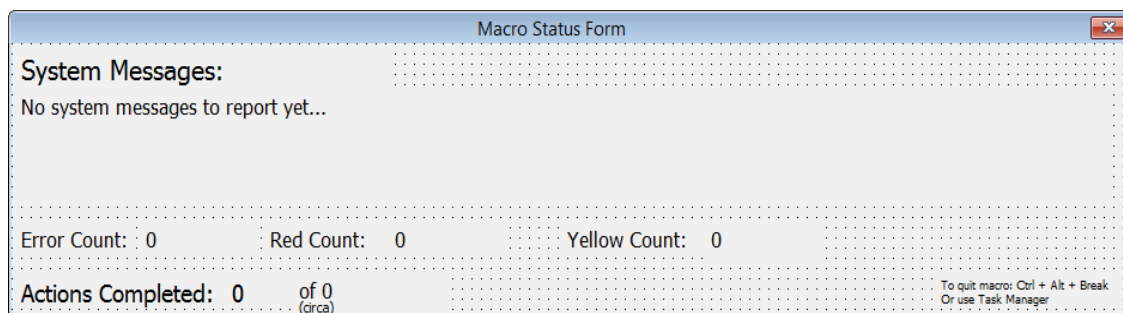
3.4.2 Komunikace programu s uživatelem

Program musí mít také možnost, jak komunikovat s uživatelem. Měl by mít možnost ukázat aktuální prováděný důležitý podprogram nebo chyby, se kterými se setkal. Tento problém jsem rozdělil do čtyř oblastí:

- Výpis nastavení programu před spuštěním (popsané v předchozí kapitole).
- Chybová hlášení – ošetření a výpis chybových stavů programu.
- Hlášení aktuálního stavu programu – pomocí samostatného stavového okna.
- Zápis průběhu programu – pro pozdější kontrolu běhu programu Zpracování.

Každá funkce a procedura programu je, podle potřeby, ohraničena kódem pro ošetření chyb. V případě chyby tudíž mohu zachytit veškeré chyby v běhu programu. Pro každou proceduru jsem si stanovil, jestli danou chybu stačí pouze vypsát, nebo jestli byla natolik závažná, že je potřeba program ukončit. Chybové hlášení pak předávám dalším metodám, které mají na starosti výpis chyb a hlášení uživateli.

Rovněž je důležité hlásit uživateli aktuální stav běhu makra. K tomu slouží samostatný formulář, který vypisuje právě prováděný krok, počty chyb a předpokládaný počet celkových kroků. Při změnách hodnot ve formuláři musím volat příkaz pro nucené překreslení formuláře, protože překreslování aplikace je zakázáno.



Obrázek č. 11: Formulář pro zobrazení průběhu zpracování (vlastní zpracování)

Uživateli by měl být po skončení programu předložen stručný popis běhu programu. Tento soupis se bude vypisovat na hlavním listu makra. Pro zpřehlednění jsem přišel s následujícím barevným značením:

- Šedá – systémová hlášení. Značí právě prováděné akce.
- Červená – značí chyby programu nebo závažné chyby v dotaznících (například chybějící jméno hodnotícího).
- Žlutá – značí možné chyby, které by měl administrátor zkontrolovat.
- Zelená – daný krok proběhl zcela v pořádku.

V této práci se už dále nebudu komunikací programu Zpracování s uživatelem zabývat. Kód pro ošetření chyb totiž musí obsahovat každá metoda, a aktualizace stavu běhu programu se bude provádět u všech stěžejních funkcí.

System messages:
Initiating... Please Wait...
Opening workbook...
Path: C:\Pathxlsm
Workbook status: green. No action required.
Path: C:\Path... .xlsm
Opening workbook...
Path: C:\Pathxlsm
No DC - is it an exception?
Sheet: Name
Workbook status: yellow. Please check data.
Path: C:\Pathxlsm
Opening workbook...
Path: C:\Pathxlsm

Obrázek č. 12: Ukázka Výpisu běhu programu (vlastní zpracování)

3.4.3 Import dat

Po spuštění makra, se spustí kód pro získání vstupních dat. Tento proces zahájí procedura ProjdiSoubory, která projde všechny sešity v nastavené složce (popřípadě podsložkách) a pro všechny sešity zavolá proceduru ZpracujSesit. Ta má za úkol otevření sešitu, předaného jako parametr, zpracování dat na listu Obecné informace, a po zpracování sešitu také hlášení stavu právě zpracovávaného sešitu. Tato metoda rovněž prochází všechny listy v sešitu, a pro každý list, kromě Obecných informací a listu s předlohou volá proceduru ZpracujList.

Funkci ZpracujList jsou vždy předané Údaje sešitu. Prvním krokem je načtení Údajů hodnoceného. Procedura poté prochází sešit po sloupcích (čtvrtletích). Pro rychlejší zpracování listu, jsem do formuláře přidal řádek s informací, jestli bylo čtvrtletí hodnocené (viz. Dodatečné úpravy formuláře). Pokud je tato hodnota nastavena jako pravdivá, tak se toto čtvrtletí začne procházet po řádcích. Zde se využije další prvek, který jsem přidal do listů hodnocení, sloupec A. V něm zapsané znaky řídí zpracování dat, přičemž prázdné hodnoty znamenají přeskočení řádku. Nahrávání dat čtvrtletí končí znakem konce vstupu. Při znaku okruhu otázek se pouze aktualizuje příslušná proměnná (skupina otázek). Při znaku otázek se aktualizují proměnné (pro otázku a hodnocení) a poté se rozhodne, do kterého listu se data uloží. O samotný zápis se stará samostatná funkce Zapis, které se zadá jméno listu a předají se jí všechny hodnoty pro zápis.

Tabulka č. 4: Význam řídících znaků Importu

Znak	Význam
A	Skupina otázek, patřící do Průměrovaných dat.
N	Skupina otázek, patřících do Neprůměrovaných dat.
S	Otázky pro číselné hodnocení.
T	Otázky pro textové hodnocení.
D	Konec vstupu.

(Vlastní zpracování)

Po skončení běhu importu je ještě vhodné spustit metodu pro vymazání duplicit ve všech tabulkách a následné uložení sešitu.

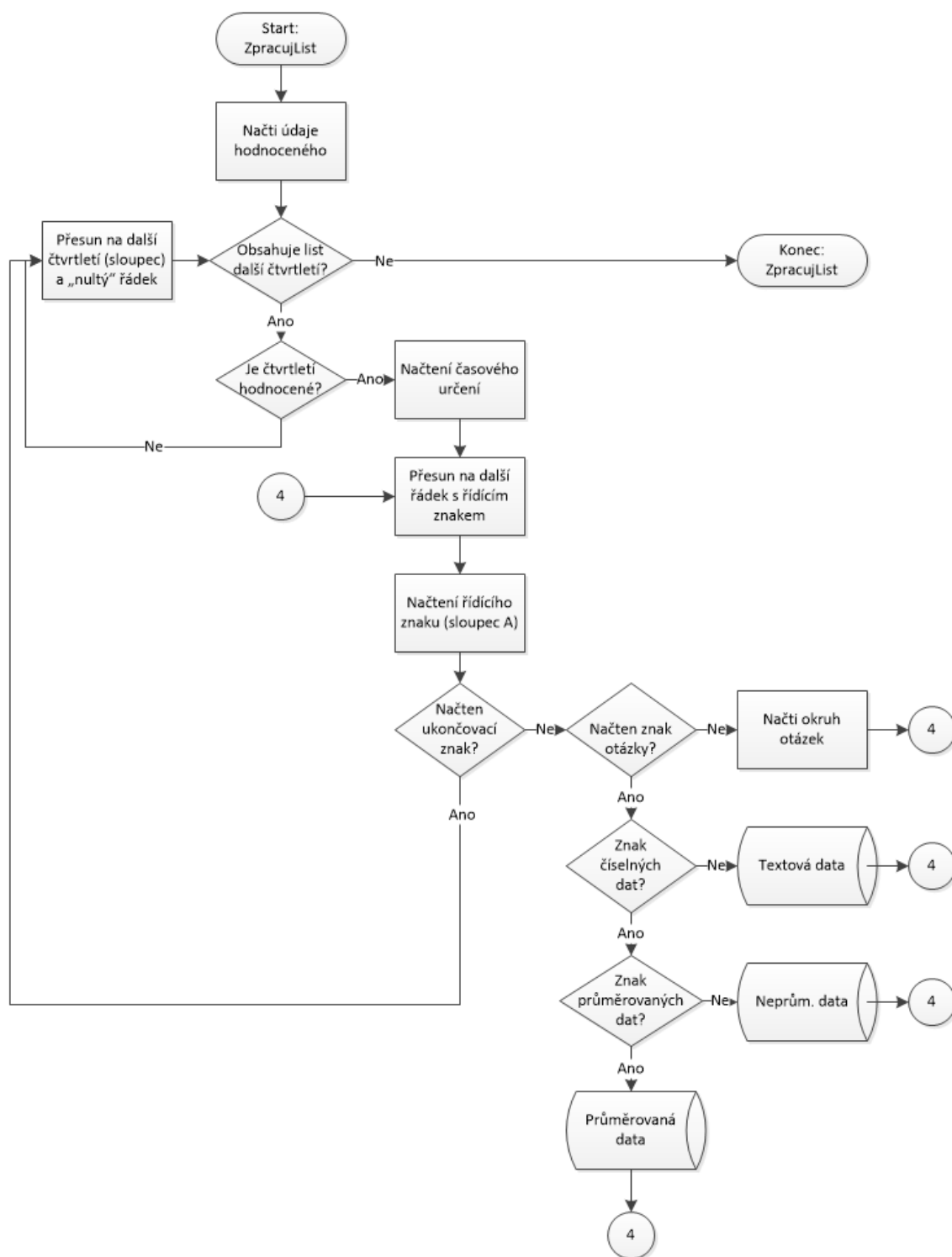


Diagram č. 3: Procedura ZpracujList (vlastní zpracování)

3.4.4 Export dat

Data získaná z dotazníků bude třeba také exportovat do výstupních souborů. Výstupní soubory musí být roztříděné tak, aby se zachovala důvěrnost dat, a také musí obsahovat různé grafy a tabulky.

V kapitole Popis řešeného problému jsem uvedl základní pravidla, podle kterých je potřeba data třídit. PPM by měl mít přístup k datům z celého oddělení a manažeri první linie potřebují znát data z jejich DC. Výstupy systému by ještě mohly zajímat portfoliové manažery a manažera Program Office. Z toho vyplývá, že bude třeba data třídit podle portfolio, DC a úrovně hodnocených (abych získal data o Project Office). V nastavení programu by mělo být možné také vybrat si, podle kterých argumentů se bude exportovat.

Export dat začne vytvořením seznamů argumentů, podle kterých se bude exportovat, načte se, podle tohoto seznamu, provede samotný export. Pro každý výstupní soubor se vytvoří kopie programu Zpracování, ve kterém se poté vyfiltrují data podle daného parametru, a vše ostatní se smaže. V každém souboru se poté budou tvořit požadované grafy a tabulky.

Třídění dat

Celá extrakce bude začínat v jediné metodě, kterou budu nazývat ZacniExport. Tato procedura si, po vytvoření výstupní složky, z nastavení zjistí, podle kterých atributů má exportovat. Pokud má provést export podle alespoň jednoho atributu, tak si program vytvoří kopii potřebných sloupců na pomocný list, a poté na nich provede odstranění duplicit. Tím získá soupis výstupních souborů, který si ještě abecedně seřadí, a podle kterého bude volat funkci pro tvorbu jednotlivých souborů (VytvorVystupniSoubor).

Metoda VytvorVystupniSoubor začne vytvořením kopie programu Zpracování. Tím dosáhnou zkopírování všech dat i procedur jediným krokem. Poté se provede vymazání listu Menu, a také se opakovaně zavolá metoda SmazPrebytecnaData pro smazání přebytečných dat v jednotlivých listech. Poté již mohu volat funkce pro tvorbu grafů v daném sešitu (ZpracujVystupniSoubor). Ta končí uložením kopie pod novým jménem a zavřením sešitu.

Procedura SmazPrebytecnaData si data v listu abecedně seřadí podle výstupního atributu, nalezne první a poslední vhodný záznam, a vše mimo tento interval (kromě hlavičky tabulky) smaže.

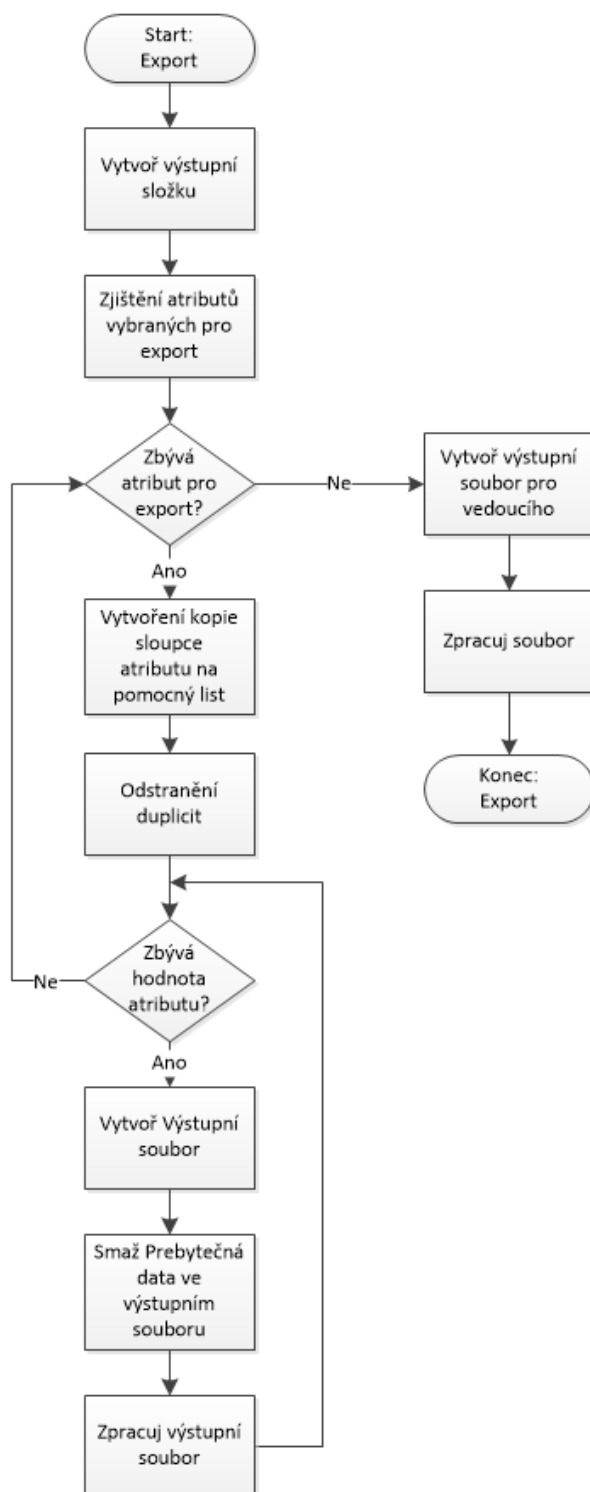


Diagram č. 4: Průběh Exportu dat (vlastní zpracování)

Získání dat pro Analýzu více hodnotitelů

V požadavcích na systém jsem definoval požadavek, podle kterého by mezi vytvořenými grafy měl být graf, který bude umožňovat porovnání hodnocení manažerů, kteří hodnotili stejného zaměstnance. Tento graf se od ostatních grafů liší tím, že bude potřebovat vlastní tabulku. Nyní se zaměřím na návrh postupu filtrování těchto dat.

Celá akce se provádí ve výstupním sešitu. Prvním krokem je seřazení tabulky průměrovaných dat podle sloupců Hodnotitele a Hodnoceného. Zde je důležité dodržet fakt, že hlavním filtrem je jméno hodnoceného manažera a druhým filtrem je jméno hodnotícího manažera. Následně se tyto sloupce zkopírují do pomocného listu pomocí metody AdvancedFilter. Jedná se o metodu oblastí, která umožňuje jejich kopírování a filtrování, a zároveň také umožňuje kopírování pouze unikátních řádků. Díky předchozímu nyní mám seznam manažerů, pro kterého se tvoří analýzu.

Nyní se v dalším sloupci pomocného listu pro všechny vyplněné řádky vytvoří funkce, která spočítá výskyty jména hodnoceného manažera z aktuálního řádku v celém sloupci. K tomu využívám běžnou funkci Excelu COUNTIF, pro kterou je parametrem „rozsah“ sloupec B a parametrem „kritérium“ je hodnota z aktuálního řádku. Po vytvoření této funkce se, na základě jejího výsledku, do pole zapisují jména hodnocených manažerů, kteří mají počet hodnocení větší, jak jedna.

Posledním krokem je, že se podle získaného pole nastaví filtr původní tabulky (metoda AutoFilter). Nakonec se „viditelná“ data zkopírují do listu Data více hodnotících.

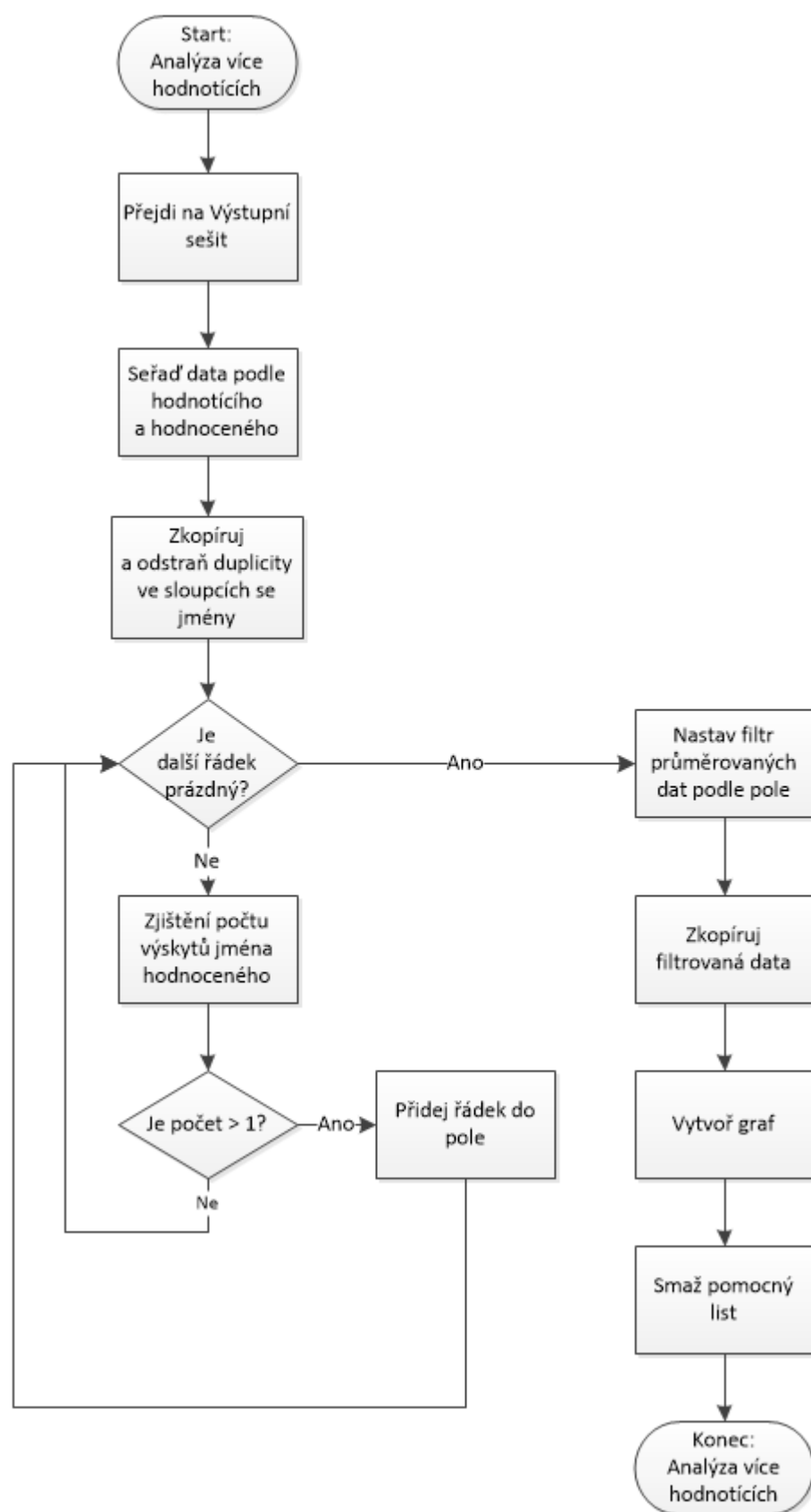


Diagram č. 5: Analýza více hodnotitelů (vlastní zpracování)

3.4.5 Tvorba grafů

Pro tvorbu grafů Excel nabízí dvě možnosti. „Obyčejné“ grafy a kontingenční grafy. „Obyčejný“ graf umožňuje zobrazení pouze jediné sady dat najednou, bez možnosti filtrování. To nabízejí právě kontingenční grafy, a proto jsem se je rozhodl použít. Díky tomu stačí tvořit pouze základní druhy grafů, které si uživatel může podle potřeby filtrovat, a program tedy nebude muset tvořit všechny možné kombinace grafů, což by bylo nepřehledné a náročné na paměť.

Tvorbu grafů a jejich nastavení jsem se rozhodl rozdělit do několika základních funkcí. První z nich VytvorKontingencniGraf má za úkol vytvoření kontingenční tabulky i grafu na samostatných listech. Vstupní parametry procedury udávají použitá data i základní nastavení. Pro filtry stránky, sloupce a řádku bude vstupem pole, obsahující názvy sloupců z datové tabulky, jejichž hodnoty chci použít pro nastavení grafu. Pro pole hodnot bude předán název sloupce, který se bude používat a argument, který bude značit, jaká funkce se má na data použít (průměr, suma). Další parametry předají požadovaný název vzniklých listů a odkaz na výstupní sešit.

Další dvě procedury (NastavGraf, NastavTypGrafu) mají za úkol dodatečné nastavení grafu. NastavGraf nastaví velikost grafu, jeho umístění na stránce a rozsah hodnot, pokud to bude potřeba. NastavTypGrafu bude, jak již napovídá název nastavovat typ grafu (sloupcový, koláčový...) a jeho rozložení.

Tvorba grafu probíhá následovně. Každý graf má svoji vlastní proceduru. Ta postupně volá procedury VytvorKontingencniGraf, NastavGraf a NastavTypGrafu. Poté se ještě tvoří tlačítka Rychlého filtrování, která popíši v další kapitole. Na závěr se graf resetuje, tak aby zobrazoval všechna data.

Rychlé filtry

Vlastníci systému také chtěli rychlejší způsob jak třídit data, než jsou klasické filtry poskytované kontingenčními grafy. Proto jsem navrhl kód, který pro každý list vytvoří sadu tlačítek, která umožní graf nastavit podle některých základních parametrů. Tato tlačítka jsem rozdělil do několika základních druhů:

- Resetovat – vrátí graf do původního stavu (zobrazí vše). Nenastavuje filtry pro rok a čtvrtletí.
- Resetovat čas – resetuje filtry pro rok a čtvrtletí.
- Přejmenovat graf – přejmenuje graf tak, aby respektovat aktuální nastavení.
- Nastavení čtvrtletí – nastaví graf pro vybrané čtvrtletí vybraného roku.
- Nastavení filtru – nastaví graf pro vybraný filtr a hodnotu.
- Rotace hodnot filtru – pro vybraný filtr zobrazí pouze další/předchozí hodnotu.

Tlačítka se tvoří dynamicky, během tvorby grafu. Nacházejí se napravo od grafu, a jsou srovnána do řádků. V prvním řádku se budou vždy nacházet tlačítka pro přejmenování, reset a reset času (pokud to je třeba). Další plochu zabírají tlačítka pro časové určení srovnaná vzestupně do řádků podle let, a poté podle čtvrtletí v řádcích. Dále se budou tvořit tlačítka pro rotaci hodnot filtru, a nakonec se vytvoří tlačítka pro nastavení filtru. Zde je nutno podotknout, že se vždy tvoří tlačítka pouze pro několik zásadních filtrů.

Každý druh tlačítek má dvě hlavní procedury. První z nich má na starosti tvorbu samotného tlačítka nebo tlačítek, jeho umístění a přiřazení funkce. Druhá procedura je volána po stisknutí tlačítka. Tvorba samotného tlačítka probíhá obecně takto:

```
Set btn = Worksheets(jmeno_listu).Buttons.Add (zleva, shora, sirka, vyska)  
btn.OnAction = " " & ActiveWorkbook.name & " '!modul.procedura" " " & _  
prvni_hodnota & " " ", "" & druha_hodnota & " " " "
```

Kód č. 3: Tvorba tlačítka a následné přiřazení volání procedury s parametry
(vlastní zpracování podle: 3)

Po vytvoření objektu tlačítka a uložení jeho reference se tlačítku nastaví jeho jméno, popis a procedura volaná po jeho stisknutí. Pro udržení unikátních jmen tlačítek ke každému jménu přidávám ještě unikátní číslo (které generuje funkce Buttons.btnIDplus). Největší problém ale nastává ve chvíli přiřazování procedury. To se děje přiřazení do vlastnosti OnAction, která umí ukládat pouze řetězce. Tyto řetězce neumí ukládat

speciální znaky a přiřazování proměnných je dosti nepřehledné. Proto byly vytvořeny dvě funkce. První, která umí zašifrovat hodnoty proměnných, aby neobsahovaly zvláštní znaky a šly tak předat funkci (a následně tyto hodnoty také dešifrovat), a druhá, která podle zadaných parametrů vytvoří řetězec pro přiřazení do vlastnosti OnAction.

Každé tlačítko, jak jsem už řekl, musí mít vlastní volanou proceduru. Všechny tyto procedury (kromě Přejmenuj) používají vlastností a procedur kontingenčních tabulek pro dosažení žádaného nastavení. Všechny tyto procedury nejdříve smažou původní nastavení a poté vytvoří nastavení nové. Tyto funkce před svým ukončením ještě zavolají přejmenování grafu. Do některých z těchto procedur je ještě v případě, že by trvaly příliš dlouho, vhodné přidat kód pro vyřazení vykreslování a přepočítávání (viz. Podpůrné metody). Těchto procedur se totiž netýká kód pro zrychlení použitý na začátku programu Zpracování.

Přejmenování grafů

Proces pro přejmenování grafu má za úkol přepsat titulek grafu tak, aby odpovídal aktuálnímu nastavení kontingenčního grafu. Jméno titulku se skládá ze dvou částí. První je obecná a popisuje obecnou funkci grafu. Druhá, proměnlivá část, poté tento název upřesňuje o nastavení filtrů. Přejmenování začíná jednoduchou funkcí, která má za úkol zavolání funkce pro získání proměnlivé části názvu (ZiskejNastaveniGrafu) a spojení obou částí titulku. Případně také zkracuje název při překročení 255 znaků (maximální délka titulku) a nakonec přepisuje titulek na nový název.

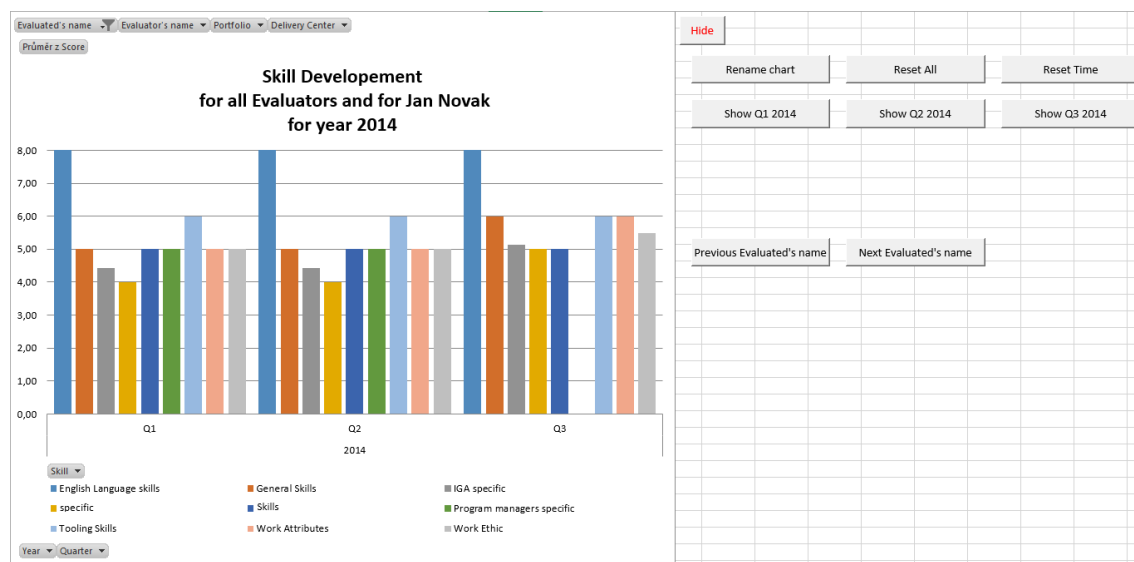
Funkce ZiskejNastaveniGrafu prochází filtry grafu a zjišťuje hodnoty, které jsou u nich nastavené. Nastavené hodnoty poté spojuje do řetězce. Pokud jsou zobrazené všechny hodnoty filtru, tak se řetězec v závěru přepíše na „all“ (vše), a pokud žádný, tak se změní na „none“ (žádný). Do druhého rozměru pole přidá název filtru (sloupce). V závěru procedury se volá funkce ZiskejTextTitulku.

Funkce ZiskejTextTitulku zpracovává pole, získané ve funkci ZiskejNastaveniGrafu a vrací již hotový text. Funkce prochází pole a podle názvu filtru určuje, jak daný řetězec upraví a zapojí do výsledku (pomocí slov jako: for, in, and).

Tvořené grafy

Výsledné grafy mohou opět rozdělit do několika základních kategorií.

- Distribuce skóre – koláčové a 100% skládané sloupkové grafy pro průměrovaná i neprůměrovaná data.
- Celkový vývoj zkušeností – grafy pro srovnání celkových průměrů hodnocení v celém oddělení.
- Celkové srovnání – tento sloupkový graf je tvořen pro portfolia, programy, DC a úrovně hodnocených. Porovnává celkové průměry dosažených hodnocení zkoumaných částí oddělení.
- Srovnání oblastí zkušeností – další sloupkový graf. Porovnává jednotlivé oblasti zkušeností pro DC, portfolia, programy a úroveň hodnocených.
- Zkušenosti zaměstnance – umožňuje porovnání jednotlivých okruhů otázek v čase (sloupkový graf).
- Celkový vývoj zaměstnance – pomocí spojnicového grafu zobrazuje průměr všech okruhů otázek zaměstnance.
- Analýza více hodnotících – umožňuje porovnat, jak různí hodnotitelé ohodnotili stejného zaměstnance (sloupkový graf).



Obrázek č. 13: Ukázka analýzy včetně rychlých filtrů (vlastní zpracování)

List „Obsah“

Problém, který vznikl vytvořením všech listů s grafy a tabulkami je poměrně velká nepřehlednost, spočívající v množství vytvořených listů. Řešením, se kterým jsem přišel, je vytvoření nového listu, který se bude chovat jako obsah nebo rozcestník. Tento list, nazvaný „Obsah“, bude obsahovat tlačítka „Zobraz“ a „Skrýj“, pro každý list sešitu. Ta budou vytvářena za běhu programu obdobně, jako tlačítka pro rychlé filtrování. Všechny ostatní listy poté mohou schovat a uživatel se po otevření výstupního sešitu zobrazí jen list Obsah, na kterém si může vybrat, která analýza ho zajímá nejvíce.

	A	B	C	D	E	F	G	H	I	J	K	L	
1		Show or Hide Charts:					Show or Hide Data Sheets:						
2													
3													
4		Score distribution pie Chart				Show	Hide	Average Data				Show	Hide
5													
6		Score distribution Chart				Show	Hide	NonAverage Data				Show	Hide
7													

Obrázek č. 14: Ukázka listu Obsah (vlastní zpracování)

3.4.6 Nastavení programu

Uživatel jakéhokoliv programu by měl mít možnost si používaný program také nastavit. To samozřejmě platí i o programu Zpracování. Nastavení tohoto programu umožňuje jednoduchý formulář. Nastavení se ukládá na listu Menu do sloupce, který je skrytý. Hlavními prvky nastavení je cesta složky pro vstup a výstup a možnost nastavení výstupních atributů.

Settings

Path to the Feedback data folder Browse

☐ Import content of subfolders?

Path to the Feedback output folder Browse

☐ Delete old data before running?

☐ Export by DC ☐ Export by Job position ☐ Export by Portfolio

Obrázek č. 15: Nastavení aplikace (vlastní zpracování)

3.4.7 Výběr formátu aplikace

Program zpracování pro zpracování dat je napsaný ve formě makra v aplikaci Microsoft Excel. Ten nabízí řadu různých souborových formátů. Do verze Microsoft Office 2007 byl hlavním (a výchozím) formátem aplikace formát xls. Jedná se o binární formát umožňující uložení uživatelských maker. Ten byl ovšem nahrazen formátemxlsx, který je formátem založeném na formátu XML, a ukládání maker neumožňuje. Současně s formátemxlsx se ale také objevily formátyxlsm (také založený na XML, ale umožňuje ukládání maker) a xlsm (binární formát, který umožňuje ukládání maker). Excel také umožňuje ukládání přímo do formátu xml a mnoha dalších typů souborů.

Pro představu jsem vytvořil jednoduché srovnání základních formátů aplikace Excel. Program Zpracování jsem nechal běžet v každém z následujících formátů třikrát. Pro každý běh programu jsem změřil čas (pomocí kódu přidaného na začátek a konec programu), a ještě jsem zapsal velikost programu po skončení běhu programu. Jako nejrychlejší se ukázal formát xlsm. Pro tento formát jsem pak ještě měřil délku běhu programu, bez vypnutého vykreslování a přepočítávání dat (řádek „xlsm bez zrychlení“).

Tabulka č. 5: Porovnání doby běhu programu při jiném nastavení

Formát	Běh 1 [s]	Běh 2 [s]	Běh 3 [s]	Průměr [s]	Velikost souboru
xls	114,18	115,16	117,40	115,58	2.25 MB
xlsm	112,18	108,94	112,27	111,13	666 KB
xlsm	108,37	103,62	105,93	105,97	415 KB
xlsm (bez zrychlení)	302,78	312,66	313,36	309,60	415 KB

(Vlastní zpracování)

Nakonec jsem se nakonec rozhodl použít formát xlsm. Umožňuje ukládání maker a hlavně v něm uložená makra jsou rychlejší. Soubor také zabírá méně místa v paměti počítače než ostatní formáty. Z předchozí tabulky je ještě vidět obrovský vliv vykreslování a automatického přepočítávání na rychlost programů v aplikaci Excel.

3.5 Návrh komunity v IBM Connections

Na komunitu v IBM Connections jsem nemusel klást mnoho požadavků. Prvním z nich je seznam jejích uživatelů. Ten je nadefinován tak, aby do ní měli přístup jen oprávnění uživatelé. To znamená administrátor, hodnotitelé a vlastníci. Do komunity dále bylo potřeba přidat komponentu Files, a do ní složku pro ukládání dotazníků, kde každý

hodnotící může číst a upravovat pouze svůj vlastní formulář. V druhé složce jsou uloženy kopie programu, dotazníků a návodů. V komunitě bylo ještě nutné uložit soubor se jmény pro aktualizaci dotazníků (přístupný všem uživatelům komunity).

Také jsem přidal složku, pro ukládání jednotlivých výstupních souborů. I zde jsem musel pamatovat na to, aby k jednotlivým souborům měli přístup pouze oprávnění vlastníci. Správcem této komunity je administrátor.

Pro zadávání požadavku pro další hodnocení lze použít modul aktivity.

3.6 Webová aplikace postavená na platformě IBM Bluemix

Při výběru řešení jsem uvedl, že navržený systém bude sloužit především pro ověření skutečných přínosů obdobného systému pro oddělení P&P. V následující kapitole tedy krátce nastíním řešení postavené na platformě IBM Bluemix, kterou bych doporučil použít při implementaci budoucí verze. Tento návrh je zamýšlený pouze pro systém hodnocení zaměstnanců, který navrhuji v této práci, ale bylo by možné do něj přidat i další funkcionality, podle potřeb oddělení.

3.6.1 Stručný popis aplikace

Pro navrhovaný systém je možné použít různé technologie, které platforma Bluemix nabízí. Ty lze rozdělit do základních kategorií:

- použité běhové prostředí,
- databáze a analytická služba,
- zabezpečení.

IBM Bluemix nabízí celou škálu běhových prostředí. Já navrhuji použití PHP nebo Node.js, přičemž v současnosti bych doporučil použití Node.js. Jedná se o modernější řešení, vhodné pro interaktivní aplikace, a které je pro platformu dostupné ve verzi od společnosti IBM, a tudíž by mělo nabízet vyšší kompatibilitu s dalšími dostupnými službami, které platforma nabízí.

Pro databáze existují dvě hlavní dostupná řešení, kterými jsou SQL a NoSQL databáze. Já bych doporučil použití SQL databáze, například službu SQL database. SQL databáze pro tento systém umožní větší konzistentnost zadávaných dat, díky svému

předdefinovanému schéma. Pro případ, že by se aplikace rozšiřovala o další funkcionalitu, by bylo vhodné uvažovat o kombinaci obou druhů databází. Výběr databází je ovšem podmíněn i výběrem služby pro analýzu získaných dat. Pro navrhovanou aplikaci, by se jako dostatečné řešení mohla ukázat služba dashDB.

Pro zabezpečení aplikace bych doporučoval použití autentizační služby Single Sign On a její propojení na autentizační služby společnosti. Aplikaci je možné rozšířit i o další služby, jako například o Data Cache nebo o Embeddable Reporting.

Návrh databáze pro ukládání hodnocení zaměstnanců

Jak jsem již řekl, pro navrhovanou aplikaci doporučuji použití relační (SQL) databáze, pro kterou navrhuji schéma, zachycené následujícím diagramem. Je ale důležité si uvědomit, že databáze slouží hlavně k analytickým účelům, a tudíž je v ní potřeba uchovávat i historické údaje.

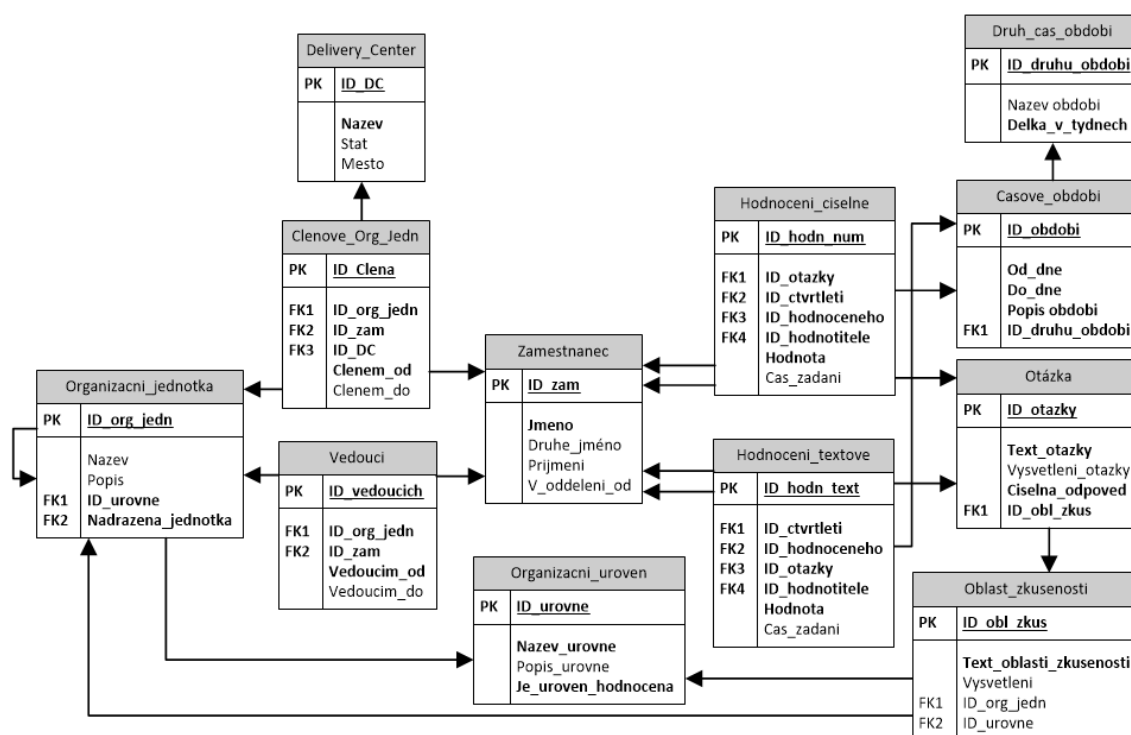


Diagram č. 6: Databázové schéma serverové aplikace Hodnocení (vlastní zpracování)

Hlavní tabulkou databáze je tabulka Zamestnanec. Její napojení na databázi zaměstnanců firmy by umožnilo snazší údržbu této databáze. Dalšími důležitými tabulkami jsou tabulky Organizacni_jednotka a Organizacni_uroven. Obě tabulky slouží k uchování celé organizační struktury oddělení a měly by obsahovat také úrovně PPM a manažerů první

linie. Tabulky Vedouci a Clenove_Org_Jedn uchovávají přiřazení zaměstnanců k organizačním jednotkám a úrovním v čase. Je důležité uvědomit si, že zaměstnanec může být zároveň vedoucím i členem nějaké organizační jednotky (manažer programu je vedoucím programu, ale členem také portfolio). Pro ukládání jednotlivých záznamů bude vhodné vytvořit příslušné podprogramy.

Druhou částí databáze je samotné hodnocení. V databázi jsou dvě tabulky určené pro ukládání hodnocení, jedna pro textové hodnocení a druhá pro číselné hodnocení. Obě tabulky jsou poté napojeny na tabulku otázek. Každá otázka musí být přiřazená k tabulce oblasti zkušeností (okruhů otázek), podle které se poté bude hodnocení seskupovat. Možnost přiřazování otázek specifických pro konkrétní organizační jednotku (například portfolio), je zajištěna připojením tabulky na tabulku organizační jednotky, a obdobně je také zajištěna možnost specifických otázek pro organizační úroveň. Obecně platné otázky tedy mají oba cizí klíče nenastavené.

Navržená databáze také umožňuje hodnocení častěji, než jednou za čtvrtletí. K tomu slouží tabulky Casove_obdobi a Druh_cas_obdobi (druh časového období). První tabulka ukládá, konkrétní časové období (například první čtvrtletí 2015) a druhá z tabulek ukládá nastavení délky časového období. K tvorbě časových období tedy bude sloužit skript, který na základě délky období v týdnech vytvoří časová období. Pro tento skript můžeme předpokládat, že čtyři týdny znamenají jeden kalendářní měsíc.

Pro celou databázi bude také potřeba vytvořit množství pohledů. Ty budou zajišťovat pro všechny zaměstnance přístup pouze k jejich datům. Pohledy tedy každému umožní prohlížení vlastní hodnocení získané od nadřízených, a vlastních hodnocení podřízených zaměstnanců.

Analytické zpracování dat

Analýza dat se může provádět pomocí jednoho z doporučených modulů – například dashDB. Pro zrychlení výpočtů je možné veškeré hodnocení (s již sečtenými hodnoceními oblastí zkušeností) ukládat do datové struktury upravené pro rychlejší výpočty. Tato možnost ale nemusí vést k výraznému zrychlení výpočtů, protože v databázi bude uložených přibližně 21 tisíc záznamů za rok, jak jsem psal v kapitole o požadavcích na systém.

Základní analýzy by bylo vhodné také předpřipravít, aby je nebylo třeba vždy tvořit ručně. Výsledné grafy by měly respektovat požadavky vedení a tudíž zahrnovat porovnání celkových zkušeností, i oblastí zkušeností pro organizačních jednotek i DC, vývoj hodnocení jednotlivých zaměstnanců, a také porovnání udělovaných hodnocení, tedy podobné grafy, jaké jsou vytvářeny v současné verzi (více v kapitole Tvorba grafů).

Aplikace Hodnocení

Samotná aplikace (nazvěme ji „Hodnocení“) se může skládat ze tří modulů. První část, bude sloužit pro udělování hodnocení, druhá pak pro prohlížení jeho výsledků. Poslední část bude přístupná pouze administrátorovi systému a bude obsahovat nastavení.

Úvodní strana systému může zobrazovat aktuální stav za posledních několik období a systémové zprávy.

Modul hodnocení po otevření hodnoticímu nabídne pracovníky, které může ohodnotit. Aplikace Hodnocení, po výběru hodnoceného, hodnotiteli předloží otázky seskupené podle oblastí zkušeností a hodnoty odpovědí z minulého hodnocení. Při zadávání číselného hodnocení musí aplikace kontrolovat, zdali zadané hodnocení odpovídá nastavené stupnici.

Modul výsledků bude nabízet veškeré analýzy dostupné pro konkrétního zaměstnance. Rovněž by bylo vhodné umožnit zaměstnanci tvorbu vlastních grafů a jejich uložení. V tomto modulu také musí být zobrazeny textová hodnocení zaměstnanců. Vhodnou funkcí, pro vedoucí oddělení, by také bylo vyhledávání zaměstnanců jménem, po kterém by se zobrazila karta zaměstnance, s grafy představujícími vývoj jeho hodnocení a výpisem textového hodnocení, které zaměstnanec obdržel.

Pro vybrané zaměstnance (PPM, manažeři první linie a administrátora) by rovněž bylo vhodné zobrazit statistiku počtu vyplněných hodnocení v aktuálním časovém období. S touto funkcí souvisí i funkce automatického upozorňování. Toho může být dosaženo pomocí automatických emailových zpráv.

3.6.2 Ekonomické zhodnocení aplikace postavené na IBM Bluemix

Cena vývoje a provozu aplikace bude záležet na rozsahu, možnostech aplikace, a zvolených službách, které IBM Bluemix nabízí. Nyní uvedu cenu doporučených služeb.

Tabulka č. 6: Ceny služeb v IBM Bluemix

Služba	Zdarma do (měsíčně)	Cena
Runtime (Node.js, PHP)	375 GB-hodin	0,07 \$ / GB-hodina
Databáze (dashDB)	1 GB paměti	50 \$ (do 20 GB paměti)
Single Sign On	10 nových uživatelů	2 \$ / uživatel

(Vlastní zpracování podle 31)

Z předchozí tabulky vyplývá, že aplikace teoreticky může běžet i zdarma. To ovšem záleží na reálné implementaci a použitých službách. Pokud bychom uvažovali, že aplikace bude potřebovat 1 GB paměti pro svůj běh a více paměti v databázi (například z důvodu rozšíření funkcionality), tak se dostáváme na cenu 74,15 \$ za měsíc.

Hlavním přínosem aplikace oproti stávajícímu řešení je možnost rozšíření funkcionality a vyšší míra automatizace oproti navrženému řešení. Aplikace může sloužit nejen vedení oddělení, ale také všem zaměstnancům, jako zdroj zpětné vazby. Aplikaci je také možné rozšířit také o další nástroje a funkce, které by oddělení mohlo v budoucnu potřebovat.

3.7 Ekonomické zhodnocení navrhovaného řešení

Pro návrh systému hodnocení zaměstnanců jsem se rozhodl použít kombinaci aplikace Microsoft Excel a podnikové sociální sítě IBM Connections. Nyní přišel čas toto řešení zhodnotit.

3.7.1 Náklady

Celý systém je postavený na softwaru, který již by na oddělení dostupný a používaný. Tím odpadají náklady spojené se zakupováním nového softwaru. Jediným nákladem tedy byly náklady na vývoj¹.

Při používání tohoto řešení také odpadají náklady spojené s hostováním aplikace nebo jejích dat, právě díky užití sítě IBM Connections. Jediným nákladem na užívání systému je čas administrátora (asi 30 – 60 minut za jedno čtvrtletí).

3.7.2 Přínosy

Bezesporu hlavním přínosem tohoto řešení je vytvoření zcela nového systému pro zkoumané oddělení. Ten umožňuje rychlé zpracování velkého množství dotazníků a vytvoření kontingenčních grafů z těchto dat. Při ruční práci by podobná analýza mohla zabrat několik dnů až týdnů „manuální“ práce.

Druhým velice důležitým přínosem je i samotné hodnocení zaměstnanců, podle zadání vedení oddělení. To může, při správném použití, motivovat mnohé zaměstnance, a pro vedení oddělení to představuje možnost zlepšení jejich povědomí o zkušenostech zaměstnanců, což by do budoucna mohlo umožnit i efektivnější řízení oddělení.

Tento systém rovněž slouží, po domluvě s vedením oddělení, jako prototyp pro otestování reálných přínosů podobného systému pro zkoumané oddělení. Pokud se systém osvědčí, tak se vedení může rozhodnout pro vytvoření jeho nové verze, postavené například na platformě IBM Bluemix. Základní vlastnosti tohoto řešení jsem nastínil v předchozí kapitole. Pokud by se tento nový systém osvědčil, tak by jej bylo možné nabízet i dalším oddělením.

¹ Tyto náklady si vedení oddělení nepřálo zveřejnit.

ZÁVĚR

Cílem této práce bylo navržení systému pro zpracování hodnocení zaměstnanců pro oddělení IGA EMEA Projects and Programmes. Po teoretickém úvodu a analýze problému, jsem na začátku třetí kapitoly popsal několik řešení, která by bylo možné použít. Systém, který jsem pro oddělení vytvořil, používá aplikace Microsoft Excel pro získání a zpracování hodnocení, a podnikové sociální síť IBM Connections pro sběr a ukládání výsledků. Oddělení toto řešení v současné době již používá. Na konci práce jsem ještě krátce popsal možné budoucí řešení postavené na cloudové platformě IBM Bluemix. Tímto považuji cíl práce za splněný.

SEZNAM POUŽITÉ LITERATURY

- (1) HRONÍK, F. *Hodnocení pracovníků*. Praha: Grada, 2006. ISBN 80-247-1458-2.
- (2) AKADEMIE ORLITA. Jak motivovat. *Orlita.cz* [online]. ©2013 [cit. 2015-05-13]. Dostupné z: <http://www.orlita.cz/services/jak-motivovat/>
- (3) MICROSOFT. VBA Language Specification Overview. *Microsoft Developer Network* [online]. ©2015 [cit. 2015-05-13]. Dostupné z: <https://msdn.microsoft.com/en-us/library/ee177270.aspx>
- (4) KRÁL, M. *Excel VBA: výukový kurz*. Brno: Computer Press, 2010. ISBN 978-80-251-2358-4.
- (5) KOCH, M. a B. NEUWIRTH. *Datové a funkční modelování*. 4. rozšířené vydání. Brno: Cerm, 2010. ISBN 978-80-214-4125-5.
- (6) WELING, L. a L. THOMSON. *PHP a MySQL: rozvoj webových aplikací*. Praha: SoftPress, 2002. ISBN 80-86497-20-8.
- (7) Introduction to NoSQL by Martin Fowler. In: *Youtube* [online]. 19.02.2013 [cit. 2015-13-05]. Dostupné z: https://www.youtube.com/watch?v=qI_g07C_Q5I. Kanál uživatele GOTO Conferences.
- (8) SODOMKA, P. a H. KLČOVÁ. *Informační systémy v podnikové praxi*. 2. aktualizované a rozšířené vydání. Brno: Computer Press, 2010. ISBN 978-80-251-2878-7.
- (9) APPRENDIA. IaaS, PaaS, SaaS (Explained and Compared). *Apprendia.com* [online]. ©2015 [cit. 2015-05-16]. Dostupné z: <http://apprendia.com/library/paas/iaas-paas-saas-explained-compared/>
- (10) ENCYCLOPÆDIA BRITANNICA. Information system. *Encyclopædia Britannica* [online]. ©2015 [cit. 2015-05-16]. Dostupné z: <http://global.britannica.com/EBchecked/topic/287895/information-system>
- (11) MICROSOFT. Overview of PivotTable and PivotChart reports. *Office* [online]. ©2015 [cit. 2015-05-16]. Dostupné z: <https://support.office.com/en-US/article/Overview-of-PivotTable-and-PivotChart-reports-00a5bf71-65cb-49f9-b321-85bb7b0b06c2>
- (12) APPEL, A. W. *A Runtime System* [online]. Princeton University: květen 1989 [16.05.2015]. CS-TR-220-89. Dostupné z: <https://users-cs.au.dk/hosc/local/LaSC-3-4-pp343-380.pdf>

- (13) TECHTERMS. API. *TechTerms.com* [online]. ©2015 [cit. 2015-05-16].
Dostupné z: <http://techterms.com/definition/api>
- (14) TECHTERMS. Framework. *TechTerms.com* [online]. ©2015
[cit. 2015-05-16]. Dostupné z: <http://techterms.com/definition/framework>
- (15) DOLÁK, O. Big data: nové způsoby zpracování a analýzy velkých objemů dat. *SystemOnline* [online]. 2011 [cit. 2015-05-16]. ISSN 1802-615X.
Dostupné z: <http://www.systemonline.cz/clanky/big-data.htm>
- (16) OMNICOM. DevOps. *Bestpractice.cz* [online]. ©2008-2015 [cit. 2015-05-16].
Dostupné z: <http://www.bestpractice.cz/cs/Best-practice/DevOps.alej>
- (17) KOCH, M. Posouzení efektivnosti informačního systému metodou HOS. *Trends Economics and Management* [online]. 2013, ročník 7, číslo 16 [cit. 2015-05-16].
ISSN 2336-6508. Dostupné z:
<http://trendy.fbm.vutbr.cz/index.php/trends/article/view/211/207>
- (18) KOCH, M. ZEFIS: posouzení efektivnosti informačních systémů. *Zefis* [online].
©2014 [cit. 2015-05-16]. Dostupné z: <http://www.zefis.cz/zefis/zefis.php>
- (19) IBM. Chronological History of IBM. *IBM.com* [online]. ©2013
[cit. 2015-01-31]. Dostupné z:
http://www-03.ibm.com/ibm/history/history/history_intro.html
- (20) *IBM* [online]. IBM, ©2013 [cit. 2015-01-31]. Dostupné z:
<http://www.ibm.com/cz/cs/>
- (21) IBM. IBM Smarter University. *IBM.com* [online]. ©2013 [cit. 2015-02-01].
Dostupné z: <http://www-05.ibm.com/cz/university/hodnoty.html>
- (22) IBM. IBM Delivery Center in Brno. *IBM.com* [online]. ©2013 [cit. 2015-01-31].
Dostupné z: http://www-05.ibm.com/employment/cz/ibm_centre_brno/
- (23) MINISTERSTVO SPRAVEDLNOSTI ČESKÉ REPUBLIKY. Výpis
z obchodního rejstříku: IBM Global Services Delivery Center Czech Republic,
s.r.o., C 39922 vedená u Krajského soudu v Brně. *eJustice* [online]. ©2012-2014
[cit. 2015-01-31]. Dostupné z: <https://or.justice.cz/ias/ui/rejstrik-firma.vysledky?subjektId=82809&typ=PLATNY>
- (24) FAYA, A. *Interview*. IBM, Technická 2995/21, Královo Pole, Brno. 15.01.2015.
- (25) NARODICKA, J. *Interview*. IBM, Technická 2995/21, Královo Pole, Brno.
12.01.2015.

- (26) LENOVO. Notebook ThinkPad T430. *Lenovo* [online]. ©2015 [cit. 2015-01-25]. Dostupné z: http://shop.lenovo.com/cz/cs/laptops/thinkpad/t-series/t430/#tab-technická_specifikace
- (27) KORBER, M. *Interview*. IBM, Technická 2995/21, Královo Pole, Brno. 02.02.2015.
- (28) OFFICE [online]. Microsoft, ©2015 [cit. 2015-01-28]. Dostupné z: <https://products.office.com/cs-CZ/>
- (29) THE NEXT WEB. Microsoft discontinues electronic forms software InfoPath, will support latest version until April 2023. *The Next Web News* [online]. ©2001–2015 [cit. 2015-01-28]. Dostupné z: <http://thenextweb.com/microsoft/2014/01/31/microsoft-discontinues-electronic-forms-software-infopath-will-support-latest-version-april-2023/>
- (30) THE APACHE SOFTWARE FOUNDATION. *Apache OpenOffice* [online]. The Apache Software Foundation, ©1999-2010 [cit. 2015-05-06]. Dostupné z: <https://www.openoffice.org/>
- (31) BRATKO, M. *Interview*. IBM, Technická 2995/21, Královo Pole, Brno. 21.01.2015.
- (32) IBM. *IBM Bluemix* [online]. IBM [cit. 2015-05-01]. Dostupné z: <https://console.ng.bluemix.net/>
- (33) THE PHP GROUP. *PHP* [online]. The PHP Group, ©2001-2015 [cit. 2015-05-05]. Dostupné z: <http://php.net/>
- (34) JOYENT. *Node.js* [online]. Joyent, ©2015 [cit. 2015-05-05]. Dostupné z: <https://nodejs.org/>
- (35) INFOWORLD. PHP vs. Node.js: An epic battle for developer mind share. *InfoWorld* [online]. ©1994 - 2015 [cit. 2015-05-17]. Dostupné z: <http://www.infoworld.com/article/2866712/php/php-vs-node-js-an-epic-battle-for-developer-mind-share.html>
- (36) ORACLE CORPORATION. *MySQL* [online]. ORACLE, ©2015 [cit. 2015-05-06]. Dostupné z: <http://www.mysql.com/>

SEZNAM POUŽITÝCH ZKRATEK

API	Application Programming Interface
DC	Delivery Center
EMEA	Europe, the Middle East and Africa
IBM	International Business Machines Corporation
IGA	IBM Global Account
P&P	IGA EMEA Projects and Programmes
PaaS	Platform as a Service
PPM	Projects and Programmes Manager
SaaS	Software as a Service
SQL	Structured Query Language
VBA	Visual Basic for Applications

SEZNAM OBRÁZKŮ

Obrázek č. 1: Příklad databázového schéma ve formě ER diagramu	16
Obrázek č. 2: Ukázka výstupu analýzy HOS 8.....	20
Obrázek č. 3: Některé symboly používané ve vývojových diagramech.....	21
Obrázek č. 4: Značky DFD podle notace Yourdon and Coad	21
Obrázek č. 5: Zjednodušené organizační schéma oddělení P&P.....	25
Obrázek č. 6: Ukázka původního formuláře	26
Obrázek č. 7: Ukázka IBM Connections	33
Obrázek č. 8: Ukázka uživatelského prostředí IPWC.....	33
Obrázek č. 9: Ukázka listu Hodnocení	46
Obrázek č. 10: Úpravy provedené na listu Hodnocení	47
Obrázek č. 11: Formulář pro zobrazení průběhu zpracování.....	51
Obrázek č. 12: Ukázka Výpisu běhu programu	51
Obrázek č. 13: Ukázka analýzy včetně rychlých filtrů	61
Obrázek č. 14: Ukázka listu Obsah.....	62
Obrázek č. 15: Nastavení aplikace.....	62

SEZNAM DIAGRAMŮ

Diagram č. 1: Diagram toku dat v navrženém systému	44
Diagram č. 2: Tok dat v programu Zpracování	49
Diagram č. 3: Procedura ZpracujList.....	53
Diagram č. 4: Průběh Exportu dat	55
Diagram č. 5: Analýza více hodnotitelů	57
Diagram č. 6: Databázové schéma serverové aplikace Hodnocení	65

SEZNAM GRAFŮ

Graf č. 1: Výsledek HOS 8 analýzy IPWC.....	34
---	----

SEZNAM TABULEK

Tabulka č. 1: Srovnání balíků Microsoft Office a Apache OpenOffice	31
Tabulka č. 2: Porovnání vlastností PHP a Node.js	37
Tabulka č. 3: Atributy, potřebné pro provedení hodnocení	44
Tabulka č. 4: Význam řídicích znaků Importu	52
Tabulka č. 5: Porovnání doby běhu programu při jiném nastavení	63
Tabulka č. 6: Ceny služeb v IBM Bluemix	68

SEZNAM KÓDU

Kód č. 1: Funkce pro kontrolu obsahu dat.....	48
Kód č. 2: Nastavení Excelu po spuštění Zpracování	49
Kód č. 3: Tvorba tlačítka a následné přiřazení volání procedury s parametry	59